Structured prediction:

\[ y: \text{DT} \quad \text{NNP} \quad \text{NNP} \quad \text{VBD} \]

\[ x: \text{The} \quad \text{European} \quad \text{Commision} \quad \text{agreed} \]

Part-of-speech tagging (POS)
Structured prediction:

\[ y: \text{DT} \quad \text{NNP} \quad \text{NNP} \quad \text{VBD} \quad x: \text{The European Commision agreed} \]

Part-of-speech tagging (POS)

\[ y: \text{O} \quad \text{B-ORG} \quad \text{I-ORG} \quad \text{O} \quad x: \text{The European Commission agreed} \]

Named-entity recognition (NER)
Structured prediction:

\[ y: \text{DT} \text{ NNP} \text{ NNP} \text{ VBD} \quad y: \ O \text{ B-ORG} \text{ I-ORG} \ O \]
\[ x: \text{The} \text{ European \ Commission \ agreed} \quad x: \text{The} \text{ European \ Commission \ agreed} \]

Part-of-speech tagging (POS)

Named-entity recognition (NER)

Current methods:

Structured models: **accurate** but **slow**

conditional random fields (CRFs) with loopy graphs, large tag sets
Structured prediction:

\[ y: \text{DT—NNP—NNP—VBD} \quad y: \text{O—B-ORG—I-ORG—O} \]
\[ x: \text{The European Commission agreed} \quad x: \text{The European Commission agreed} \]

Part-of-speech tagging (POS)

Named-entity recognition (NER)

**Current methods:**

Structured models: **accurate but slow**

\[
\begin{array}{c}
\quad \quad \quad \quad \quad \\
\quad \quad \quad \quad \quad \\
\quad \quad \quad \quad \quad \\
\quad \quad \quad \quad \quad \\
\end{array}
\]
conditional random fields (CRFs) with loopy graphs, large tag sets

Independent models: **less accurate but fast**

\[
\begin{array}{c}
\quad \quad \quad \quad \quad \\
\quad \quad \quad \quad \quad \\
\quad \quad \quad \quad \quad \\
\quad \quad \quad \quad \quad \\
\end{array}
\]
independent logistic regressions (ILRs)
Structured prediction:

\[ y: \text{DT NNP NNP VBD} \quad y: \text{O B-ORG I-ORG O} \]
\[ x: \text{The European Commission agreed} \quad x: \text{The European Commission agreed} \]

Part-of-speech tagging (POS)  
Named-entity recognition (NER)

Current methods:

Structured models: accurate but slow
- conditional random fields (CRFs) with loopy graphs, large tag sets

Independent models: less accurate but fast
- independent logistic regressions (ILRs)

Our goal:

\[ \text{transfer predictive power} \quad \text{accurate and fast at test time} \]
Structured prediction:

\[ y: \text{DT} \quad \text{NNP} \quad \text{NNP} \quad \text{VBD} \quad y: \text{O} \quad \text{B-ORG} \quad \text{I-ORG} \quad \text{O} \]

\[ x: \text{The} \quad \text{European} \quad \text{Commission} \quad \text{agreed} \quad x: \text{The} \quad \text{European} \quad \text{Commission} \quad \text{agreed} \]

Part-of-speech tagging (POS)      Named-entity recognition (NER)

Current methods:

Structured models: accurate but slow
- conditional random fields (CRFs) with loopy graphs, large tag sets

Independent models: less accurate but fast
- independent logistic regressions (ILRs)

Our goal:

\[ \text{transfer} \quad \text{predictive power} \quad \text{accurate and fast at test time} \]

Questions: are independent models...
- ...expressive enough (approximation error)?
Structured prediction:

\[
\begin{align*}
y &: \text{DT} --- \text{NNP} --- \text{NNP} --- \text{VBD} \\
x &: \text{The European Commision agreed}
\end{align*}
\]

Part-of-speech tagging (POS)

Named-entity recognition (NER)

Current methods:

Structured models: accurate but slow

conditioned random fields (CRFs) with loopy graphs, large tag sets

Independent models: less accurate but fast

independent logistic regressions (ILRs)

Our goal:

\[
\begin{align*}
\text{transfer} & \quad \text{predictive power} \\
\text{accurate and fast} & \text{ at test time}
\end{align*}
\]

Questions: are independent models...

• ...expressive enough \text{(approximation error)}?  
• ...easy to learn \text{(estimation error)}?
Some empirical motivation

CRF($f_1$)

POS: 95.0%
NER: 75.3%

$f_1$: words/prefixes/suffixes/forms
Some empirical motivation

CRF($f_1$)  
POS: 95.0%  
NER: 75.3%  

ILR($f_1$)  
POS: 91.7%  
NER: 69.1%  

$f_1$: words/prefixes/suffixes/forms
Some empirical motivation

CRF($f_1$)  POS: 95.0%  
NER: 75.3%

$f_1$: words/prefixes/suffixes/forms

ILR($f_1$)  POS: 91.7%  
NER: 69.1%

ILR($f_2$)  POS: 94.4%  
NER: 66.2%

$f_2$: $f_1$ applied to larger radius

f_1$ applied to larger radius
Some empirical motivation

**CRF**($f_1$)  
POS: 95.0%  
NER: 75.3%

$f_1$: words/prefixes/suffixes/forms

**ILR**($f_1$)  
POS: 91.7%  
NER: 69.1%

$f_2$: $f_1$ applied to larger radius

**ILR**($f_2$)  
POS: 94.4%  
NER: 66.2%
Some empirical motivation

CRF($f_1$)  
POS: 95.0%  
NER: 75.3%

$f_1$: words/prefixes/suffixes/forms

$\text{ILR}(f_1)$  
POS: 91.7%  
NER: 69.1%

$\text{ILR}(f_2)$  
POS: 94.4%  
NER: 66.2%

$\text{ILR}(f_2)$ [compiled]  
POS: 95.0%  
NER: 72.7%

$f_2$: $f_1$ applied to larger radius
Some empirical motivation

CRF($f_1$)  
- POS: 95.0%
- NER: 75.3%

$f_1$: words/prefixes/suffixes/forms

$ILR(f_1)$  
- POS: 91.7%
- NER: 69.1%

$ILR(f_2)$  
- POS: 94.4%
- NER: 66.2%

$ILR(f_2)$ [compiled]  
- POS: 95.0%
- NER: 72.7%

Structure compilation: reduces the gap between the ILR and CRF
Analysis of structure compilation

CRF\( (f_1) \) → auto-labeled data → ILR\( (f_2) \)
Analysis of structure compilation

Goal: analyze risk of final compiled $\text{ILR}(f_2)$
Analysis of structure compilation

Goal: analyze risk of final compiled $\text{ILR}(f_2)$

Decomposition of errors:

Approximation error: best loss of model (with infinite data)
Analysis of structure compilation

Goal: analyze risk of final compiled $\text{ILR}(f_2)$

Decomposition of errors:

Approximation error: best loss of model (with infinite data)
Estimation error: suboptimality due to finite data
Approximation/estimation errors for ILR

CRF($f_1$) → auto-labeled data → ILR($f_2$)
Approximation/estimation errors for ILR

\[ \text{CRF}(f_1) \rightarrow \text{auto-labeled data} \rightarrow \text{ILR}(f_2) \]

\( p_c : \text{CRF}(f_1) \) trained on labeled data
Approximation/estimation errors for ILR

\[ p_C: \text{CRF}(f_1) \text{ trained on labeled data} \]

\[ p_1: \text{ILR}(f_2) \text{ trained on } m \text{ auto-labeled examples} \]
Approximation/estimation errors for ILR

\[ \text{CRF}(f_1) \]
\[ \text{ILR}(f_2) \]

\( p_C: \text{CRF}(f_1) \text{ trained on labeled data} \)

\( p_I: \text{ILR}(f_2) \text{ trained on } m \text{ auto-labeled examples} \)

\( p_I^*: \text{ILR}(f_2) \text{ trained on infinite auto-labeled data} \)
Approximation/estimation errors for ILR

\[ \text{CRF}(f_1) \longrightarrow \text{auto-labeled data} \longrightarrow \text{ILR}(f_2) \]

\( p_C: \text{CRF}(f_1) \) trained on labeled data

\( p_I: \text{ILR}(f_2) \) trained on \( m \) auto-labeled examples

\( p_I^*: \text{ILR}(f_2) \) trained on infinite auto-labeled data

\[
\text{KL} (p_C \parallel p_I) = \text{KL} (p_C \parallel p_I^*) + (\text{KL} (p_C \parallel p_I) - \text{KL} (p_C \parallel p_I^*))
\]

approx. error

estimation error
Approximation/estimation errors for ILR

\[ \text{CRF}(f_1) \quad \longrightarrow \quad \text{auto-labeled data} \quad \longrightarrow \quad \text{ILR}(f_2) \]

\[ p_c: \text{CRF}(f_1) \text{ trained on labeled data} \]
\[ p_i: \text{ILR}(f_2) \text{ trained on } m \text{ auto-labeled examples} \]
\[ p_i*: \text{ILR}(f_2) \text{ trained on infinite auto-labeled data} \]

\[ \text{KL} (p_c \parallel p_i) = \text{KL} (p_c \parallel p_i^*) + \left( \text{KL} (p_c \parallel p_i) - \text{KL} (p_c \parallel p_i^*) \right) \]

Estimation error:

\[ \text{Expected value} = \frac{\# \text{ features}}{m} + o \left( \frac{1}{m} \right) \rightarrow 0 \quad [\text{Liang & Jordan, 2008}] \]
Approximation/estimation errors for ILR

\[ \text{CRF}(f_1) \quad \text{ILR}(f_2) \]

\( p_C : \text{CRF}(f_1) \) trained on labeled data
\( p_I : \text{ILR}(f_2) \) trained on \( m \) auto-labeled examples
\( p_{I^*} : \text{ILR}(f_2) \) trained on infinite auto-labeled data

\[
\text{KL} (p_C \| p_I) = \underbrace{\text{KL} (p_C \| p_{I^*})}_{\text{approx. error}} + \left( \text{KL} (p_C \| p_I) - \text{KL} (p_C \| p_{I^*}) \right)_{\text{estimation error}}
\]

**Estimation error:**

Expected value \( \frac{\# \text{ features}}{m} + o\left(\frac{1}{m}\right) \rightarrow 0 \) [Liang & Jordan, 2008]
Structured compilation can eliminate this error
Approximation/estimation errors for ILR

\[ \text{CRF}(f_1) \rightarrow \text{auto-labeled data} \rightarrow \text{ILR}(f_2) \]

\( p_c : \) CRF\((f_1)\) trained on labeled data

\( p_i : \) ILR\((f_2)\) trained on \(m\) auto-labeled examples

\( p_i^* : \) ILR\((f_2)\) trained on infinite auto-labeled data

\[ \text{KL} (p_c \| p_i) = \text{KL} (p_c \| p_i^*) + \left( \text{KL} (p_c \| p_i) - \text{KL} (p_c \| p_i^*) \right) \]

Estimation error:

Expected value = \(\frac{\# \text{ features}}{m} + o\left(\frac{1}{m}\right) \rightarrow 0\) [Liang & Jordan, 2008]

Structured compilation can eliminate this error

Approximation error: next...
Decomposition of approximation error

\[ \text{CRF}(f_1): p_c \]

\[ \text{ILR}(f_2): p_{i^*} \]
Decomposition of approximation error

CRF\( (f_1) : p_c \)

marginalized CRF\( (f_1) : p_{MC} \)

ILR\( (f_2) : p_{i^*} \)
Decomposition of approximation error

CRF\((f_1)\): \(p_C\)

coherence

marginalized CRF\((f_1)\): \(p_{MC}\)

ILR\((f_2)\): \(p_{1^*}\)
Decomposition of approximation error

CRF($f_1$): $p_C$

marginalized CRF($f_1$): $p_{MC}$

ILR($f_\infty$): $p_{A^*}$

ILR($f_2$): $p_{1^*}$
Decomposition of approximation error

CRF\( (f_1) \): \( p_c \)

marginalized CRF\( (f_1) \): \( p_{MC} \)

ILR\( (f_{\infty}) \): \( p_{A^*} \)

ILR\( (f_2) \): \( p_{1^*} \)

coherence

nonlinearities

global information
Decomposition of approximation error

\[
\text{CRF}(f_1): p_C \qquad \text{coherence}
\]

\[
\text{marginalized CRF}(f_1): p_{MC} \quad \Sigma \quad \Sigma \quad \Sigma \quad \text{nonlinearities}
\]

\[
\text{ILR}(f_\infty): p_{A^*} \quad \text{global information}
\]

\[
\text{ILR}(f_2): p_{I^*}
\]

**Theorem:**

\[
\text{KL} (p_C \| p_{I^*}) = \text{KL} (p_C \| p_{MC}) + \text{KL} (p_{MC} \| p_{A^*}) + \text{KL} (p_{A^*} \| p_{I^*})
\]
Decomposition of approximation error

CRF($f_1$): $p_C$

marginalized CRF($f_1$): $p_{MC}$

ILR($f_\infty$): $p_{A^*}$

ILR($f_2$): $p_{1^*}$

Theorem:

$$KL(p_C \parallel p_{1^*}) = KL(p_C \parallel p_{MC}) + KL(p_{MC} \parallel p_{A^*}) + KL(p_{A^*} \parallel p_{1^*})$$

Proof:

Generalized Pythagorean identity for KL-divergence
Approximation error: coherence

CRF: $p_c$

marginalized CRF: $p_{MC}$

Coherence $= \text{KL} (p_c \| p_{MC})$: importance of making joint predictions
Approximation error: coherence

CRF: $p_C$

marginalized CRF: $p_{MC}$

Coherence $= \text{KL} (p_C || p_{MC})$: importance of making joint predictions

For a chain CRF:

coherence $= \text{sum of mutual information along the edges}$
Approximation error: coherence

CRF: $p_C$

marginalized CRF: $p_{MC}$

Coherence $= \text{KL} (p_C \mid\mid p_{MC})$:
importance of making joint predictions

For a chain CRF:
coherence $= \text{sum of mutual information along the edges}$

<table>
<thead>
<tr>
<th>POS</th>
<th>NER</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.003</td>
<td>0.009</td>
</tr>
</tbody>
</table>
Approximation error: coherence

\[ \text{Coherence} = \text{KL} (p_c \| p_{MC}) : \]

importance of making joint predictions

For a chain CRF:

coherence = sum of mutual information along the edges

<table>
<thead>
<tr>
<th>POS</th>
<th>NER</th>
</tr>
</thead>
<tbody>
<tr>
<td>Coherence</td>
<td>0.003</td>
</tr>
<tr>
<td>Change in accuracy</td>
<td>95.0% ⇒ 95.0%</td>
</tr>
</tbody>
</table>
Approximation error: coherence

Coherence = \( KL(p_c || p_{MC}) \):
importance of making joint predictions

For a chain CRF:
coherence = sum of mutual information along the edges

<table>
<thead>
<tr>
<th></th>
<th>POS</th>
<th>NER</th>
</tr>
</thead>
<tbody>
<tr>
<td>Coherence</td>
<td>0.003</td>
<td>0.009</td>
</tr>
<tr>
<td>Change in accuracy</td>
<td>95.0% ⇒ 95.0%</td>
<td>76.3% ⇒ 76.0%</td>
</tr>
</tbody>
</table>

**Coherence is not a huge concern** (for these applications)
Approximation error: nonlinearities

marginalized CRF: $p_{MC}$

$\text{ILR}(f_{\infty}): p_{A^*}$

Nonlinearities $= \text{KL} (p_{MC} \parallel p_{A^*})$: importance of combining features in a nonlinear way
Approximation error: nonlinearities

marginalized CRF: $p_{MC}$

$\text{ILR}(f_\infty): p_{A^*}$

Nonlinearities = $\text{KL} (p_{MC} \parallel p_{A^*})$:

importance of combining features in a nonlinear way

NER experiment:

Train a truncated CRF, so that both the truncated CRF (nonlinear) and the ILR (linear) use the same features
Approximation error: nonlinearities

marginalized CRF: $p_{MC}$

ILR($f_{\infty}$): $p_{A^*}$

Nonlinearities = $\text{KL} (p_{MC} \parallel p_{A^*})$:
importance of combining features in a nonlinear way

NER experiment:
Train a truncated CRF, so that both the truncated CRF (nonlinear) and the ILR (linear) use the same features

<table>
<thead>
<tr>
<th>Truncated CRF</th>
<th>ILR($f_2$)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Accuracy</td>
<td>76.0%</td>
</tr>
</tbody>
</table>
Approximation error: nonlinearities

marginalized CRF: $p_{MC}$

ILR($f_\infty$): $p_{A^*}$

Nonlinearities = $\text{KL} (p_{MC} \parallel p_{A^*})$:

importance of combining features in a nonlinear way

NER experiment:

Train a truncated CRF, so that both the truncated CRF (nonlinear) and the ILR (linear) use the same features

<table>
<thead>
<tr>
<th>Truncated CRF</th>
<th>ILR($f_2$)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Accuracy</td>
<td>76.0%</td>
</tr>
</tbody>
</table>

Nonlinearities play an important role
Approximation error: global information

ILR\left(f_\infty\right): p_{A^*}

ILR\left(f_2\right): p_{I^*}

Global information = KL \left(p_{A^*} \parallel p_{I^*}\right):

importance of using features on distant parts of the input
Approximation error: global information

\[ \text{ILR}(f_{\infty}): p_{A^*} \]
\[ \text{ILR}(f_2): p_{I^*} \]

Global information = KL \((p_{A^*} \mid\mid p_{I^*})\):

importance of using features on distant parts of the input

NER experiment:

Compare truncated CRF with marginalized CRF (they differ only in the features used)
Approximation error: global information

\[ \text{ILR}(f_\infty): p_{A^*} \]

\[ \text{ILR}(f_2): p_{I^*} \]

Global information \( = \text{KL}(p_{A^*} \parallel p_{I^*}) \):

importance of using features on distant parts of the input

NER experiment:

Compare truncated CRF with marginalized CRF (they differ only in the features used)

<table>
<thead>
<tr>
<th></th>
<th>Marginalized CRF</th>
<th>Truncated CRF</th>
</tr>
</thead>
<tbody>
<tr>
<td>Accuracy</td>
<td>76.0%</td>
<td>76.0%</td>
</tr>
</tbody>
</table>
Approximation error: global information

\[
\text{ILR}(f_\infty): \ p_{A^*}
\]

\[
\text{ILR}(f_2): \ p_{I^*}
\]

Global information = \( \text{KL} (p_{A^*} \| p_{I^*}) \):

importance of using features on distant parts of the input

NER experiment:

Compare truncated CRF with marginalized CRF (they differ only in the features used)

<table>
<thead>
<tr>
<th></th>
<th>Marginalized CRF</th>
<th>Truncated CRF</th>
</tr>
</thead>
<tbody>
<tr>
<td>Accuracy</td>
<td>76.0%</td>
<td>76.0%</td>
</tr>
</tbody>
</table>

Distant information is not essential (for these applications)
Structure compilation for parsing

S

NP
DT The
NN cat
VBD ate

NP
DT a
JJ tasty
NN fish

10
Structure compilation for parsing

Sentence length: $\ell$
Number of grammar symbols: $K$
Number of grammar rules: $G \gg \ell, K$
Structure compilation for parsing

Sentence length: $\ell$
Number of grammar symbols: $K$
Number of grammar rules: $G \gg \ell, K$

Parse time/sentence
$O(\ell^3 G)$

Structured model:
Standard dynamic program for context-free grammars
Structure compilation for parsing

Sentence length: $\ell$
Number of grammar symbols: $K$
Number of grammar rules: $G \gg \ell, K$

Structured model:

Standard dynamic program for context-free grammars

Independent model:

For each of $O(\ell^2)$ spans:

Make a soft prediction of whether it’s a constituent
(features: words/tags/prefixes/suffixes on entire span)
Structure compilation for parsing

Sentence length: \( \ell \)
Number of grammar symbols: \( K \)
Number of grammar rules: \( G \gg \ell, K \)

Structured model:
Standard dynamic program for context-free grammars
\[ O(\ell^3 G) \]

Independent model:
For each of \( O(\ell^2) \) spans:
Make a soft prediction of whether it’s a constituent
(features: words/tags/prefixes/suffixes on entire span)
Run a dynamic program to choose the best tree
\[ O(\ell^3 + K \ell^2) \]
Parsing results

4K labeled sentences
Parsing results

4K labeled sentences

40K labeled sentences
Structure is important in parsing

Need richer features or nonlinearities for the independent model to catch up
Summary of structure compilation

Motivation: want fast CRF-level accuracy at test time
Summary of structure compilation

Motivation: want fast CRF-level accuracy at test time
Summary of structure compilation

Motivation: want fast CRF-level accuracy at test time

CRF($f_1$)
computationally complex

ILR($f_2$)
computationally simple

labeled data

auto-labeled data
Summary of structure compilation

Motivation: want fast CRF-level accuracy at test time

\[
\text{CRF}(f_1) \quad \text{ILR}(f_2)
\]

- CRF: computationally complex, statistically simple
- ILR: computationally simple, statistically complex

Estimation error: structure compilation can easily drive it to 0
Summary of structure compilation

Motivation: want fast CRF-level accuracy at test time

CRF($f_1$)
- computationally complex
- statistically simple
- very expressive

ILR($f_2$)
- computationally simple
- statistically complex
- not as expressive

Estimation error: structure compilation can easily drive it to 0

Approximation error: advantages of CRF over ILR
Summary of structure compilation

Motivation: want fast CRF-level accuracy at test time

CRF($f_1$)
- computationally complex
- statistically simple
- very expressive

ILR($f_2$)
- computationally simple
- statistically complex
- not as expressive

Estimation error: structure compilation can easily drive it to 0

Approximation error: advantages of CRF over ILR
- ILR needs rich features to compensate
Summary of structure compilation

Motivation: want fast CRF-level accuracy at test time

CRF($f_1$)
- computationally complex
- statistically simple
- very expressive

ILR($f_2$)
- computationally simple
- statistically complex
- not as expressive

Estimation error: structure compilation can easily drive it to 0

Approximation error: advantages of CRF over ILR
- ILR needs rich features to compensate
- CRF’s nonlinearities are important