Lectures on Natural Language Processing

11. HMMs and PCFGs

Karl Stratos

Structured Prediction

Each input x has a set of valid "structures" $\mathcal{Y}(x)$ as labels.

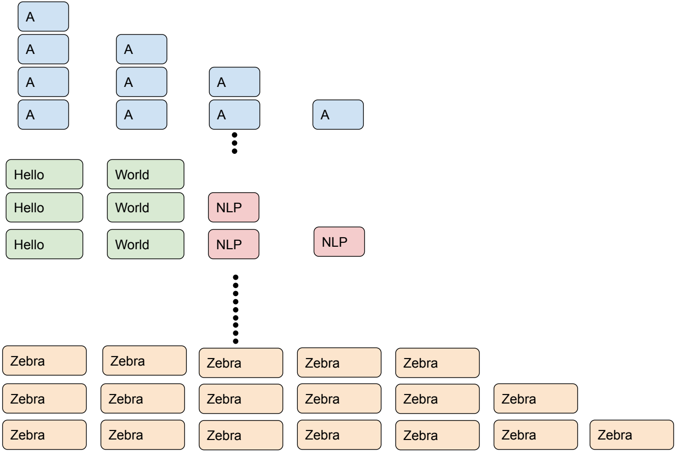
$$\max_{y \in \mathcal{Y}(x)} \textbf{score}(x, y) \qquad \textbf{(decoding/search problem)}$$

$$\sum_{y \in \mathcal{Y}(x)} \textbf{score}(x, y) \qquad \textbf{(marginalization problem)}$$

Why can't we just calculate the max/sum?

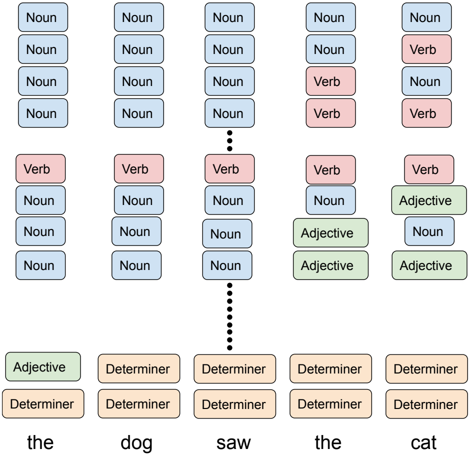
Example: Translation

$$x \in \mathcal{V}_{\text{src}}^{T},\ \mathcal{Y}(x) = \cup_{t \leq T'} \mathcal{V}_{\text{tgt}}^{t}$$



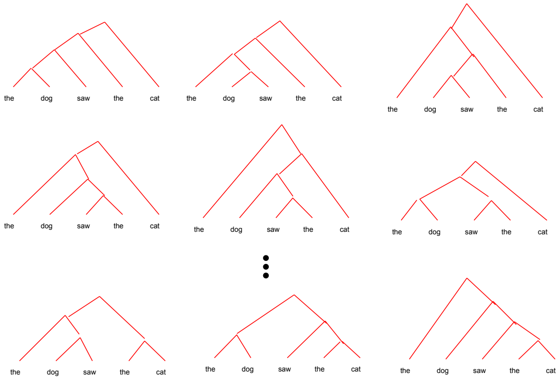
Example: Sequence Labeling/Tagging

$$x \in \mathcal{V}^T,\ \mathcal{Y}(x) = \mathcal{Y}^T$$



Example: Parsing

$x \in \mathcal{V}^T$, $\mathcal{Y}(x) =$ all possible binary trees over T tokens



Catalyn numbers: $1, 1, 2, 5, 14, 42, 132, 429, 1430, 4862, 16796, 58786, \ldots$

Beyond Beam Search

- In general, no way to avoid exhaustive search: approximation by beam search
- Can we do *exact search* by making certain assumptions?
- **Yes.** Key assumption: conditional independence
- Focus: tagging and parsing, with two different types of graphical models
  1. **Directed graphical models (aka., Bayesian networks).**
     - Hidden Markov models (HMMs) for tagging
     - Probabilistic context-free grammars (PCFGs) for parsing
  2. **Undirected graphical models (aka., Markov/conditional random fields).**
     - CRF tagger and parser
- All structured prediction models can be "neuralized" (i.e., parameterize the base score function with a neural network).

Tagging Example: Part-Of-Speech (POS) Tagging

- Given a sentence, output a sequence of POS tags.
- Ambiguity: a word can have many possible POS tags

  `the/DT man/NN saw/VBD the/DT cut/NN`

  `the/DT saw/NN cut/VBD the/DT man/NN`
- Definition of POS tags in Penn Treebank (English)

Tag	Description	Example	Tag	Description	Example	Tag	Description	Example
CC	coordinating conjunction	*and, but, or*	PDT	predeterminer	*all, both*	VBP	verb non-3sg present	*eat*
CD	cardinal number	*one, two*	POS	possessive ending	*'s*	VBZ	verb 3sg pres	*eats*
DT	determiner	*a, the*	PRP	personal pronoun	*I, you, he*	WDT	wh-determ.	*which, that*
EX	existential 'there'	*there*	PRP$	possess. pronoun	*your, one's*	WP	wh-pronoun	*what, who*
FW	foreign word	*mea culpa*	RB	adverb	*quickly*	WP$	wh-possess.	*whose*
IN	preposition/ subordin-conj	*of, in, by*	RBR	comparative adverb	*faster*	WRB	wh-adverb	*how, where*
JJ	adjective	*yellow*	RBS	superlatv. adverb	*fastest*	$	dollar sign	*$*
JJR	comparative adj	*bigger*	RP	particle	*up, off*	#	pound sign	*#*
JJS	superlative adj	*wildest*	SYM	symbol	*+,%, &*	“	left quote	‘ or “
LS	list item marker	*1, 2, One*	TO	“to”	*to*	”	right quote	’ or ”
MD	modal	*can, should*	UH	interjection	*ah, oops*	(	left paren	[, (, {, <
NN	sing or mass noun	*llama*	VB	verb base form	*eat*	)	right paren	],), }, >
NNS	noun, plural	*llamas*	VBD	verb past tense	*ate*	,	comma	,
NNP	proper noun, sing.	*IBM*	VBG	verb gerund	*eating*	.	sent-end punc	. ! ?
NNPS	proper noun, plu.	*Carolinas*	VBN	verb past part.	*eaten*	:	sent-mid punc	: ; ... – -

Figure 8.1 Penn Treebank part-of-speech tags (including punctuation).

[45 tags]

(Marcus et al., 1993)

Other definitions: universal tagset (12 tags, language agnostic)

Tagging Example: Named-Entity Recognition (NER)

- **Task.** Given a sentence, identify and label all spans that are "named entities"

$$\ldots\ \overbrace{\texttt{John Smith}}^{\text{PER}}\ \texttt{works at}\ \overbrace{\texttt{New York Times}}^{\text{ORG}}\ \ldots$$

- **Reduction to tagging.** "Linearize" labeled spans into a label sequence using "BIO" scheme

`John/B-PER Smith/I-PER works/O at/O New/B-ORG York/I-ORG Times/I-ORG`

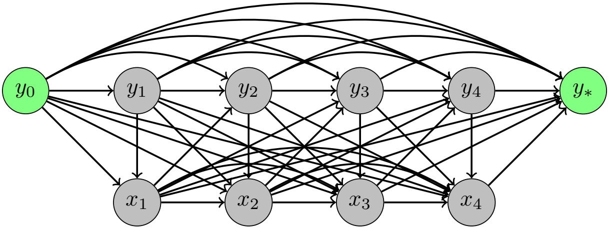
Number of tagging labels: $2 \times$ number of entity types $+ 1$

```
West    B-MISC
Indian  I-MISC
all-rounder     O
Phil    B-PER
Simmons I-PER
took    O
four    O
for     O
38      O
on      O
Friday  O
as      O
Leicestershire  B-ORG
beat    O
Somerset        B-ORG
by      O
```

CoNLL 2003 dataset, 4 entity types (PER, ORG, LOC, MISC)

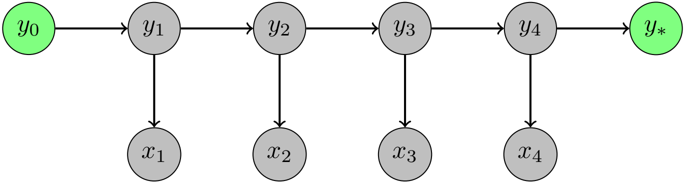
Probabilistic Generative Tagger

Observations $x_1 \dots x_T \in \mathcal{V}$, labels $y_1 \dots y_T \in \mathcal{Y}$ (start/end $y_0, y_* \in \mathcal{Y}$)



$$
\begin{aligned}
p(x_1 \dots x_T,\ y_1 \dots y_T) &= p(y_1 | y_0) && \text{(start with } y_1\text{)} \\
&\times p(x_1 | y_0\ y_1) && \text{(emit } x_1\text{)} \\
&\times p(y_2 | x_1, y_0\ y_1) && \text{(transition to } y_2\text{)} \\
&\times p(x_2 | x_1, y_0\ y_1\ y_2) && \text{(emit } x_2\text{)} \\
&\times \cdots \\
&\times p(y_T | x_1 \dots x_{T-1}, y_1 \dots y_{T-1}) && \text{(transition to } y_T\text{)} \\
&\times p(x_T | x_1 \dots x_{T-1}, y_1 \dots y_T) && \text{(emit } x_T\text{)} \\
&\times p(y_* | x_1 \dots x_T, y_1 \dots y_T) && \text{(end)}
\end{aligned}
$$

Hidden Markov Model (HMM)



$$p(x_1 \dots x_T,\ y_1 \dots y_T) = \prod_{t=1}^{T} \underbrace{\tau(y_t|y_{t-1})}_{\text{transition prob}} \times \underbrace{o(x_t|y_t)}_{\text{emission prob}} \times \tau(y_*|y_T)$$

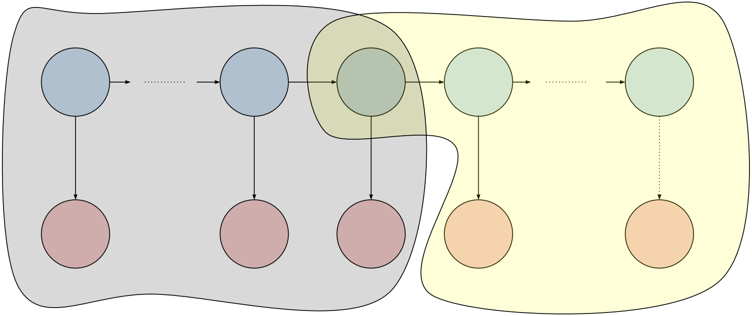
Markov assumptions. At any step t,

$$p(y_t|x_1 \dots x_{t-1}, y_1 \dots y_{t-1}) = \tau(y_t|y_{t-1})$$
$$p(x_t|x_1 \dots x_{t-1}, y_1 \dots y_t) = o(x_t|y_t)$$

Are these reasonable assumptions for tagging?

Conditional Independence Under HMMs

The future is independent of the past conditioning on the current label.



Verify that under an HMM, at any step t:

$$p(x_1 \ldots x_T,\ y_1 \ldots y_T) = p(x_1 \ldots x_t,\ y_1 \ldots y_t) \times p(x_{t+1} \ldots x_T,\ y_{t+1} \ldots y_T | y_t)$$

Supervised Learning of HMMs

Given N tagged sequences, maximum likelihood estimate (MLE)

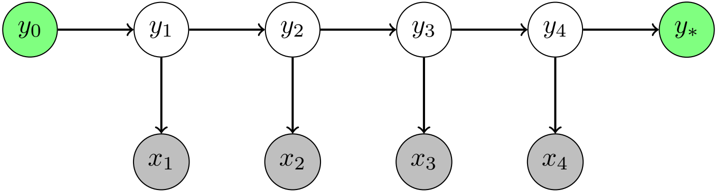
$$\tau^\star, o^\star = \arg\max_{\tau \in \mathcal{T},\, o \in \mathcal{O}} \sum_{i=1}^{N} \log p(x_1^{(i)} \ldots x_{T_i}^{(i)},\ y_1^{(i)} \ldots y_{T_i}^{(i)})$$

Constrained optimization: Lagrangian relaxation shows

$$\tau^\star(y'|y) \propto \text{(number of times } y \text{ transitions to } y' \text{ in data)}$$
$$o^\star(x|y) \propto \text{(number of times } y \text{ emits } x \text{ in data)}$$

E.g., given $\{(a\ b, A\ B), (z\ c, A\ C)\}$, we estimate $\tau^\star(y_*|B) = 1$, $\tau^\star(B|A) = \tau^\star(C|A) = \frac{1}{2}$, $o^\star(b|B) = 1$, $o^\star(a|A) = o^\star(z|A) = \frac{1}{2}$, etc.

The Marginalization Problem



Given HMM parameters and an observed sequence $x_1 \ldots x_T$ (without labels), what is the probability of that sequence under the HMM?

$$\sum_{y_1 \ldots y_T \in \mathcal{Y}} p(x_1 \ldots x_T,\ y_1 \ldots y_T)$$

Number of possible label sequences: exponential in length

Forward Algorithm

Dynamic programming: Given $x_1 \dots x_T$, we fill out a table $\alpha \in \mathbb{R}^{T \times |\mathcal{Y}|}$ left-to-right where

$$\alpha(t, y) = \sum_{y_1 \dots y_t \in \mathcal{Y}:\ y_t = y} p(x_1 \dots x_t, y_1 \dots y_t)$$

Base case?

$$\alpha(1, y) =$$

Forward Algorithm

Dynamic programming: Given $x_1 \ldots x_T$, we fill out a table $\alpha \in \mathbb{R}^{T \times |\mathcal{Y}|}$ left-to-right where

$$\alpha(t, y) = \sum_{y_1 \ldots y_t \in \mathcal{Y}:\ y_t = y} p(x_1 \ldots x_t, y_1 \ldots y_t)$$

Base case?

$$\alpha(1, y) = \tau(y|y_0) \times o(x_1|y)$$

Forward Algorithm: Main Body ($t > 1$)

$$
\begin{aligned}
\alpha(t, y') &= \sum_{y_1 \ldots y_t :\; y_t = y'} p(x_1 \ldots x_t,\; y_1 \ldots y_t) \\
&= \sum_{y_1 \ldots y_{t-1}} p(x_1 \ldots x_t,\; y_1 \ldots y_{t-1}\; y') \\
&= \sum_{y_1 \ldots y_{t-1}} p(x_1 \ldots x_{t-1},\; y_1 \ldots y_{t-1}) \times p(y' | x_1 \ldots x_{t-1},\; y_1 \ldots y_{t-1}) \\
&\qquad \times p(x_t | x_1 \ldots x_{t-1},\; y_1 \ldots y_{t-1}\; y')
\end{aligned}
$$

Forward Algorithm: Main Body ($t > 1$)

$$
\begin{aligned}
\alpha(t, y') &= \sum_{y_1 \ldots y_t:\ y_t = y'} p(x_1 \ldots x_t,\ y_1 \ldots y_t) \\
&= \sum_{y_1 \ldots y_{t-1}} p(x_1 \ldots x_t,\ y_1 \ldots y_{t-1}\ y') \\
&= \sum_{y_1 \ldots y_{t-1}} p(x_1 \ldots x_{t-1},\ y_1 \ldots y_{t-1}) \times p(y' | x_1 \ldots x_{t-1},\ y_1 \ldots y_{t-1}) \\
&\qquad \times p(x_t | x_1 \ldots x_{t-1},\ y_1 \ldots y_{t-1}\ y') \\
&:= \sum_{y_1 \ldots y_{t-1}} p(x_1 \ldots x_{t-1},\ y_1 \ldots y_{t-1}) \times \tau(y' | y_{t-1}) \times o(x_t | y')
\end{aligned}
$$

Forward Algorithm: Main Body ($t > 1$)

$$
\begin{aligned}
\alpha(t, y') &= \sum_{y_1 \ldots y_t:\ y_t = y'} p(x_1 \ldots x_t,\ y_1 \ldots y_t) \\
&= \sum_{y_1 \ldots y_{t-1}} p(x_1 \ldots x_t,\ y_1 \ldots y_{t-1}\ y') \\
&= \sum_{y_1 \ldots y_{t-1}} p(x_1 \ldots x_{t-1},\ y_1 \ldots y_{t-1}) \times p(y' | x_1 \ldots x_{t-1},\ y_1 \ldots y_{t-1}) \\
&\qquad \times p(x_t | x_1 \ldots x_{t-1},\ y_1 \ldots y_{t-1}\ y') \\
&:= \sum_{y_1 \ldots y_{t-1}} p(x_1 \ldots x_{t-1},\ y_1 \ldots y_{t-1}) \times \tau(y' | y_{t-1}) \times o(x_t | y') \\
&= \sum_{y} \sum_{y_1 \ldots y_{t-2}} p(x_1 \ldots x_{t-1}, y_1 \ldots y_{t-2}\ y) \times \tau(y' | y) \times o(x_t | y')
\end{aligned}
$$

Forward Algorithm: Main Body ($t > 1$)

$$
\begin{aligned}
\alpha(t, y') &= \sum_{y_1 \ldots y_t:\ y_t = y'} p(x_1 \ldots x_t,\ y_1 \ldots y_t) \\
&= \sum_{y_1 \ldots y_{t-1}} p(x_1 \ldots x_t,\ y_1 \ldots y_{t-1}\ y') \\
&= \sum_{y_1 \ldots y_{t-1}} p(x_1 \ldots x_{t-1},\ y_1 \ldots y_{t-1}) \times p(y' | x_1 \ldots x_{t-1},\ y_1 \ldots y_{t-1}) \\
&\qquad\qquad \times p(x_t | x_1 \ldots x_{t-1},\ y_1 \ldots y_{t-1}\ y') \\
&:= \sum_{y_1 \ldots y_{t-1}} p(x_1 \ldots x_{t-1},\ y_1 \ldots y_{t-1}) \times \tau(y' | y_{t-1}) \times o(x_t | y') \\
&= \sum_{y} \sum_{y_1 \ldots y_{t-2}} p(x_1 \ldots x_{t-1}, y_1 \ldots y_{t-2}\ y) \times \tau(y' | y) \times o(x_t | y') \\
&= \sum_{y} \alpha(t-1, y) \times \tau(y' | y) \times o(x_t | y')
\end{aligned}
$$

Forward Algorithm for HMMs: Summary

Input: HMM parameters (t, o), observed sequence $x_1 \ldots x_T \in \mathcal{V}$
Output: $\alpha(t, y) = \sum_{y_1 \ldots y_t \in \mathcal{Y}:\ y_t = y} p(x_1 \ldots x_t, y_1 \ldots y_t)$ for all $t = 1 \ldots T$ and $y \in \mathcal{Y}$

1. For all $y \in \mathcal{Y}$, compute

$$\alpha(1, y) = \tau(y|y_0) \times o(x_1|y)$$

2. For $t = 2 \ldots T$:

   2.1 For all $y' \in \mathcal{Y}$, compute

$$\alpha(t, y') = \sum_{y \in \mathcal{Y}} \alpha(t-1, y) \times \tau(y'|y) \times o(x_t|y')$$

Runtime?

Aside: Forward Algorithm in Matrix Form

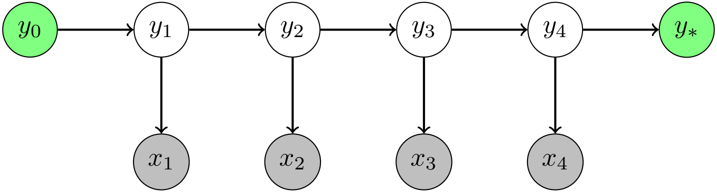
- Organize HMM probabilities in matrix form
  - Emission matrix: $O \in \mathbb{R}^{|\mathcal{V}| \times |\mathcal{Y}|}$ where $O_{x,y} = o(x|y)$
  - Transition matrix: $T \in \mathbb{R}^{|\mathcal{Y}| \times |\mathcal{Y}|}$ where $T_{y',y} = t(y'|y)$
- Forward algorithm

$$p(x_1 \ldots x_T) = \underbrace{\tau_\infty^\top}_{1 \times |\mathcal{Y}|} \underbrace{\mathrm{diag}(O_{x_T})}_{|\mathcal{Y}| \times |\mathcal{Y}|} \underbrace{T}_{|\mathcal{Y}| \times |\mathcal{Y}|} \cdots \underbrace{\mathrm{diag}(O_{x_1})}_{|\mathcal{Y}| \times |\mathcal{Y}|} \underbrace{\tau_0}_{|\mathcal{Y}| \times 1}$$

  $O_x \in \mathbb{R}^{|\mathcal{Y}|}$ is row x of O, $[\tau_0]_y = t(y|y_0)$, $[\tau_\infty]_y = t(y_*|y)$
- Stepwise marginalization as matrix-matrix product

$$\sum_{y \in \mathcal{Y}} \alpha(t-1, y) \times t(y'|y) \times o(x_t|y')$$

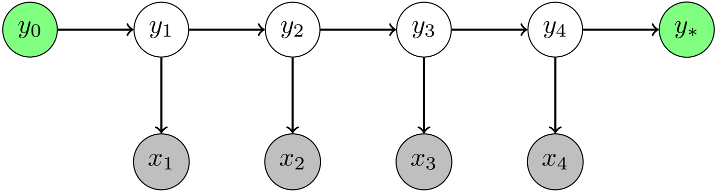
Marginalization: Solved by the Forward Algorithm



$$\sum_{y_1 \ldots y_T \in \mathcal{Y}} p(x_1 \ldots x_T,\; y_1 \ldots y_T)$$

$$= \sum_{y \in \mathcal{Y}} \alpha(T, y) \times \tau(y_* | y)$$

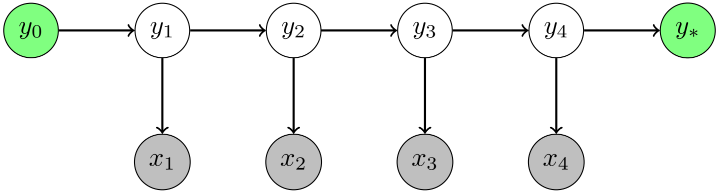
The Decoding Problem



Given HMM parameters and an observed sequence $x_1 \ldots x_T$, what is the most likely tag sequence under the HMM?

$$y_1^\star \ldots y_T^\star = \arg\max_{y_1 \ldots y_T \in \mathcal{Y}} \; p(y_1 \ldots y_T \mid x_1 \ldots x_T)$$

The Decoding Problem



Given HMM parameters and an observed sequence $x_1 \ldots x_T$, what is the most likely tag sequence under the HMM?

$$
\begin{aligned}
y_1^\star \ldots y_T^\star &= \underset{y_1 \ldots y_T \in \mathcal{Y}}{\arg\max}\; p(y_1 \ldots y_T \mid x_1 \ldots x_T) \\
&= \underset{y_1 \ldots y_T \in \mathcal{Y}}{\arg\max}\; p(x_1 \ldots x_T, y_1 \ldots y_T)
\end{aligned}
$$

Viterbi Algorithm

Given $x_1 \dots x_T$, we fill out a table $\pi \in \mathbb{R}^{T \times |\mathcal{Y}|}$ left-to-right where

$$\pi(t, y) = \max_{y_1 \dots y_t \in \mathcal{Y}:\ y_t = y} p(x_1 \dots x_t, y_1 \dots y_t)$$

Same as forward except we switch sum with max! Base case?

$$\pi(1, y) = \tau(y|y_0) \times o(x_1|y)$$

Main body? Verify that

$$\pi(t, y') = \max_{y \in \mathcal{Y}} \pi(t-1, y) \times \tau(y'|y) \times o(x_t|y')$$

Backtracking for Viterbi

- Using Viterbi, we compute the *probability* of $x_1 \ldots x_T$ and the most likely tag sequence in $O(T\,|\mathcal{Y}|^2)$ by

$$p(x_1 \ldots x_T,\ y_1^\star \ldots y_T^\star) = \max_{y \in \mathcal{Y}}\ \ \pi(T, y) \times \tau(y_*|y)$$

- Well, how do we get the actual tag sequence $y_1^\star \ldots y_T^\star$?

Backtracking for Viterbi

- Using Viterbi, we compute the *probability* of $x_1 \ldots x_T$ and the most likely tag sequence in $O(T\,|\mathcal{Y}|^2)$ by

$$p(x_1 \ldots x_T,\ y_1^\star \ldots y_T^\star) = \max_{y \in \mathcal{Y}}\ \pi(T, y) \times \tau(y_* | y)$$

- Well, how do we get the actual tag sequence $y_1^\star \ldots y_T^\star$?
- Keep an additional back-pointer to record the path:

$$\textbf{bp}(t, y') = \arg\max_{y \in \mathcal{Y}}\ \pi(t-1, y) \times \tau(y'|y) \times o(x_t|y')$$

No additional computational overhead

Summary of Viterbi Decoding

Input: HMM parameters (t, o), observed sequence $x_1 \ldots x_T \in \mathcal{V}$
Output: $\pi(t, y) = \max_{y_1 \ldots y_t \in \mathcal{Y}:\, y_t = y}\ p(x_1 \ldots x_t, y_1 \ldots y_t)$ for all $t = 1 \ldots T$ and $y \in \mathcal{Y}$, corresponding back-pointer **bp**, most likely tag sequence $y_1^\star \ldots y_T^\star$

1. For all $y \in \mathcal{Y}$, compute

$$\pi(1, y) = \tau(y|y_0) \times o(x_1|y)$$

2. For $t = 2 \ldots T$:

    2.1 For all $y' \in \mathcal{Y}$, compute

$$\pi(t, y') = \max_{y \in \mathcal{Y}}\ \pi(t-1, y) \times \tau(y'|y) \times o(x_t|y')$$

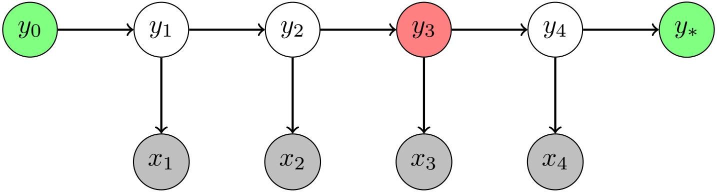
$$\mathbf{bp}(t, y') = \arg\max_{y \in \mathcal{Y}}\ \pi(t-1, y) \times \tau(y'|y) \times o(x_t|y')$$

3. Extract $y_1^\star \ldots y_T^\star$ as follows:

$$y_T^\star = \arg\max_{y \in \mathcal{Y}}\ \pi(T, y) \times \tau(y_*|y)$$

$$y_{t-1}^\star = \mathbf{bp}(t, y_t^\star) \qquad \text{for } t = T \ldots 2$$

Alternative Decoding Method: Marginal Decoding

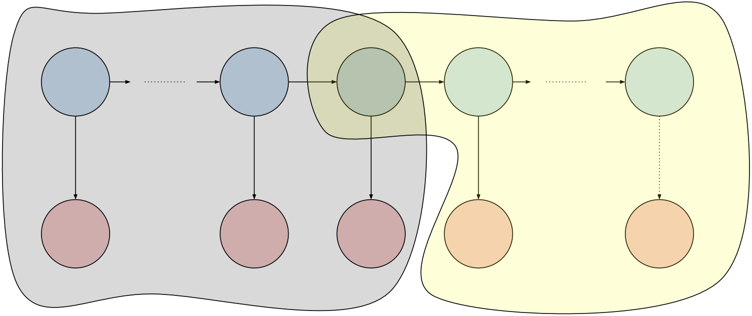


Given HMM parameters and an observed sequence $x_1 \dots x_T$, what is the most likely tag *at each step* under the HMM?

$$y_t^\star = \arg\max_{y \in \mathcal{Y}} \underbrace{\sum_{y_1 \dots y_T \in \mathcal{Y}:\, y_t = y} p(x_1 \dots x_T,\ y_1 \dots y_T)}_{\text{“marginal” } \mu(t,y)}$$

Different from Viterbi decoding, optimizes per-position accuracy

Decomposition of Marginal Under HMMs



$$\mu(t,y) = \sum_{y_1 \ldots y_T \in \mathcal{Y}:\; y_t = y} p(x_1 \ldots x_T,\; y_1 \ldots y_T)$$

$$= \sum_{y_1 \ldots y_T \in \mathcal{Y}:\; y_t = y} p(x_1 \ldots x_t,\; y_1 \ldots y_t) \times p(x_{t+1} \ldots x_T,\; y_{t+1} \ldots y_T | y_t)$$

$$= \underbrace{\sum_{y_1 \ldots y_t \in \mathcal{Y}:\; y_t = y} p(x_1 \ldots x_t,\; y_1 \ldots y_t)}_{\text{Where have we seen this before?}} \times \underbrace{\sum_{y_t \ldots y_T \in \mathcal{Y}:\; y_t = y} p(x_{t+1} \ldots x_T,\; y_{t+1} \ldots y_T)}_{\text{How do we calculate this?}}$$

Backward Algorithm

Given $x_1 \dots x_T$, we fill out a table $\beta \in \mathbb{R}^{T \times |\mathcal{Y}|}$ *right-to-left* where

$$\beta(t, y) = \sum_{y_t \dots y_T \in \mathcal{Y}:\ y_t = y} p(x_{t+1} \dots x_T, y_{t+1} \dots y_T)$$

Base case?

$$\beta(T, y) =$$

Backward Algorithm

Given $x_1 \ldots x_T$, we fill out a table $\beta \in \mathbb{R}^{T \times |\mathcal{Y}|}$ *right-to-left* where

$$\beta(t, y) = \sum_{y_t \ldots y_T \in \mathcal{Y}:\ y_t = y} p(x_{t+1} \ldots x_T, y_{t+1} \ldots y_T)$$

Base case?

$$\beta(T, y) = \tau(y_* | y)$$

Backward Algorithm: Main Body ($t < T$)

$$\begin{aligned}
\beta(t, y) &= \sum_{y_t \ldots y_T \in \mathcal{Y}:\ y_t = y} p(x_{t+1} \ldots x_T, y_{t+1} \ldots y_T) \\
&= \sum_{y_{t+1} \ldots y_T \in \mathcal{Y}} p(x_{t+1} \ldots x_T, y_{t+1} \ldots y_T | y_t = y) \\
&= \sum_{y_{t+1} \ldots y_T \in \mathcal{Y}} \tau(y_{t+1} | y) \times o(x_{t+1} | y_{t+1}) \times p(x_{t+2} \ldots x_T, y_{t+2} \ldots y_T | y_{t+1}) \\
&= \sum_{y'} \sum_{y_{t+1} \ldots y_T \in \mathcal{Y}} \tau(y' | y) \times o(x_{t+1} | y') \times p(x_{t+2} \ldots x_T, y_{t+2} \ldots y_T | y_{t+1} = y') \\
&= \sum_{y'} \tau(y' | y) \times o(x_{t+1} | y') \times \sum_{y_{t+1} \ldots y_T \in \mathcal{Y}} p(x_{t+2} \ldots x_T, y_{t+2} \ldots y_T | y_{t+1} = y') \\
&= \sum_{y'} \tau(y' | y) \times o(x_{t+1} | y') \times \beta(t+1, y')
\end{aligned}$$

Summary of Marginal Decoding

Input: HMM parameters, observed sequence $x_1 \dots x_T \in \mathcal{V}$
Output: Max-marginal tags $y_1^\star \dots y_T^\star \in \mathcal{Y}$

1. Run forward algorithm to compute for all t, y $O(T\,|\mathcal{Y}|^2)$

$$\alpha(t,y) = \sum_{y_1 \dots y_t \in \mathcal{Y}:\ y_t = y} p(x_1 \dots x_t, y_1 \dots y_t)$$

2. Run backward algorithm to compute for all t, y $O(T\,|\mathcal{Y}|^2)$

$$\beta(t,y) = \sum_{y_t \dots y_T \in \mathcal{Y}:\ y_t = y} p(x_{t+1} \dots x_T, y_{t+1} \dots y_T)$$

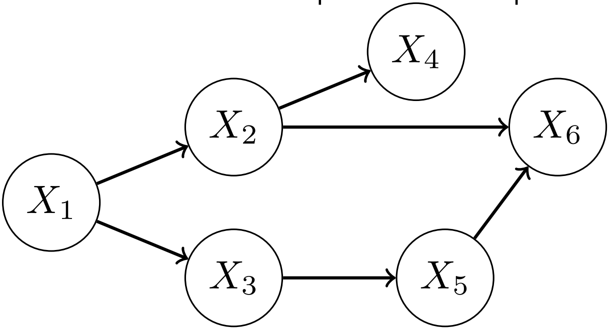
3. For each position $t = 1 \dots T$, predict as the label of x_t

$$y_t^\star = \arg\max_{y \in \mathcal{Y}}\ \alpha(t,y) \times \beta(t,y)$$

Directed Graphical Models (DGMs)

HMM is a special case of a **directed graphical model** (DGM), aka. **Bayesian network** (Bayes net)

- Graph representing a joint distribution, (lack of) directed edges encode conditional independence assumptions

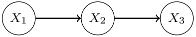


(example by David Blei)

$$\Pr(X_1, X_2, X_3, X_4, X_5, X_6)$$
$$= \Pr(X_1)\Pr(X_2|X_1)\Pr(X_3|X_1)\Pr(X_4|X_2)\Pr(X_5|X_3)\Pr(X_6|X_2, X_5)$$

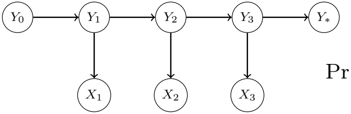
Examples of DGM

- n-gram language models with Markov order 1



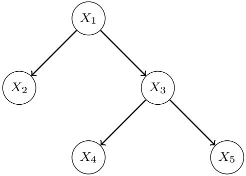
$$\Pr(X) = \Pr(X_1)\Pr(X_2|X_1)\Pr(X_3|X_2)$$

- HMMs



$$\Pr(X,Y) = \prod_{t=1}^{3}\Pr(Y_t|Y_{t-1})\Pr(X_t|Y_t)\Pr(Y_*|Y_3)$$

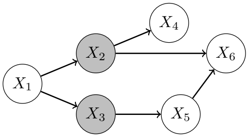
- Trees



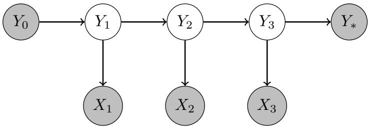
$$\Pr(X) = \Pr(X_1)\Pr(X_2|X_1)\Pr(X_3|X_1)\Pr(X_4|X_3)\Pr(X_5|X_3)$$

Observed vs Unobserved Variables in DGM

Calculate various probabilities in the presence of observed variables



$\Pr(X_2 = x_2, X_3 = x_3)$



$\max_{y_1,y_2,y_3} \Pr(X_1 = x_1, X_2 = x_2, X_3 = x_3, Y_1 = y_1, Y_2 = y_2, Y_3 = y_3)$

Conditional independence assumptions in DGMs make efficient marginalization/inference possible

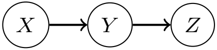
- Recall: X, Z independent $(X \perp\!\!\!\perp Z)$ conditioned on Y iff

$$\Pr(X = x|Y = y, Z = z) = \Pr(X = x|Y = y)$$

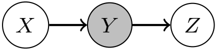
for all values of x, y, z (equiv. $p(x, y|z) = p(x|z)p(y|z)$)

Rules of Conditional Independence in DGMs

- The future is independent of the past given the present (Markov assumption)

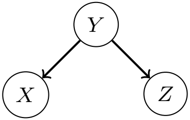


$X \not\perp\!\!\!\perp Z$

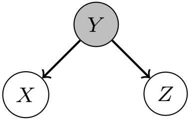


$X \perp\!\!\!\perp Z \mid Y$

- Children are independent of each other given their parent

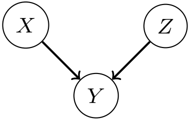


$X \not\perp\!\!\!\perp Z$

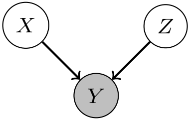


$X \perp\!\!\!\perp Z \mid Y$

- Causes are independent, but become dependent if effect is observed



$X \perp\!\!\!\perp Z$

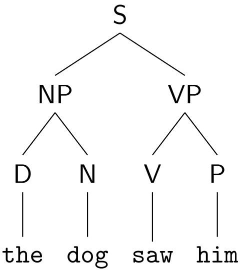


$X \not\perp\!\!\!\perp Z \mid Y$

- Exercise: Verify independence claims mathematically, and think of examples for non-independence claims

Constituency Parsing and PCFGs

Constituency tree for the sentence "the dog saw him"



Probabilistic context-free grammars (PCFGs): generative model of parses defining

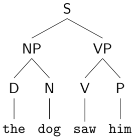
$p($  $) = \prod_{\text{rule}\in}$  $q(\text{rule})$

PCFG: Definition

A PCFG is a tuple $G = (N, \Sigma, R, S, q)$ where

- N: non-terminal symbols (constituents)
- Σ: terminal symbols (words)
- R: rules of form $X \to Y_1 \ldots Y_m$ where $X \in N, Y_i \in N \cup \Sigma$
- $S \in N$: start symbol
- q: rule probability $q(\alpha \to \beta) \geq 0$ for every rule $\alpha \to \beta \in R$ such that $\sum_\beta q(X \to \beta) = 1$ for any $X \in N$

A tree is generated top-down by starting from S and sampling rule expansions $\alpha \to \beta$ left-to-right, depth-first.

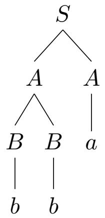


$$\equiv \quad \begin{array}{c} \text{S} \to \text{NP VP}, \text{NP} \to \text{D N} \\ \text{D} \to \text{the}, \text{N} \to \text{dog}, \text{VP} \to \text{V P} \\ \text{V} \to \text{saw}, \text{P} \to \text{him} \end{array}$$

Example PCFG

$$N = \{S, A, B\}$$
$$\Sigma = \{a, b\}$$
$$R = \{S \to A\ A,\ S \to A\ B,\ A \to B\ B,\ A \to a,\ B \to b\}$$

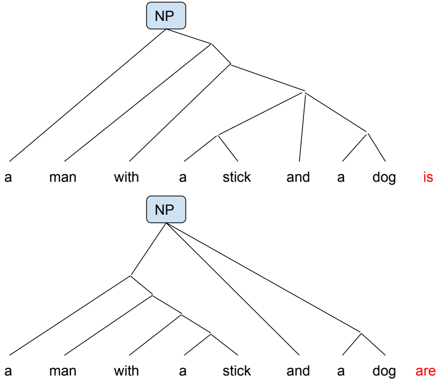
$$q(S \to A\ A) = 0.4 \qquad q(S \to A\ B) = 0.6$$
$$q(A \to B\ B) = 0.1 \qquad q(A \to a) = 0.9$$
$$q(B \to b) = 1$$



$p($  $) = q(S \to A\ A)q(A \to B\ B)q(B \to b)^2 q(A \to a) = 0.036$

Conditional Independence Under PCFGs

A subtree is independent of everything above, given its root.



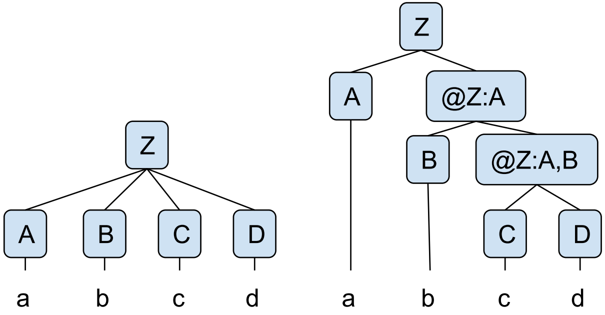
Too strong for natural language syntax: how should we parse "a man with a stick and a dog", given that it's a noun phrase?

Chomsky Normal Form (CNF)

WLOG, we can assume that a PCFG is in CNF, meaning every rule $\alpha \to \beta \in R$ is either

1. (Binary) $X \to Y\ Z$ where $X, Y, Z \in N$
2. (Unary) $X \to x$ where $X \in N$, $x \in \Sigma$

Possible to convert between a PCFG and its CNF version by introducing additional non-terminals



Estimating a PCFG from a Treebank

Given trees  $^{(1)}$ ...  $^{(N)}$ in the training data

- N: all non-terminal symbols (constituents) seen in the data
- Σ: all terminal symbols (words) seen in the data
- R: all rules seen in the data
- $S \in N$: special start symbol (if the data does not already have it, add it to every tree)
- q: Maximum-likelihood estimate (MLE) given by

$$q(\alpha \to \beta) = \frac{\textbf{count}(\alpha \to \beta)}{\sum_{\beta} \textbf{count}(\alpha \to \beta)}$$

If we see $A \to B\ C$ 3 times and A 10 times, than $q(A \to B\ C) = 0.3$

Aside: Improper PCFG

$$A \to A\ A \qquad \text{with probability } \gamma$$

$$A \to a \qquad \text{with probability } 1 - \gamma$$

Lemma. Define

$$S^* = \lim_{h \to \infty} \left(\sum_{t:\ \text{height}(\quad) \le h} p(\quad) \right)$$

If $\gamma > 0.5$, then $S^* < 1$.

- Total probability of parses is less than one! Happens because some trees grow forever.
- Fortunately, an MLE from a finite treebank is never improper (aka. "tight") (Chi and Geman, 2015)

Marginalization and Inference

GEN$(x_1 \dots x_T)$ denotes the set of all valid  's for $x_1 \dots x_T$ under the considered PCFG.

1. What is the probability of $x_1 \dots x_T$ under a PCFG?

$\sum$ $p($  $)$

 $\in$**GEN**$(x_1 ... x_T)$

2. What is the most likely tree of $x_1 \dots x_T$ under a PCFG?

$\arg\max$ $p($  $)$

 $\in$**GEN**$(x_1 ... x_T)$

Inside Algorithm

- The **inside algorithm** computes, bottom up, for all $1 \leq i \leq j \leq T$, for all $X \in N$,

$$\alpha(i,j,X) = \sum_{\in \textsf{\textbf{GEN}}(x_i \ldots x_j):\ \mathrm{root}(\quad)=X} p(\quad)$$

  We will see that computing each $\alpha(i,j,X)$ takes $O(T\,|R|)$ time.
  - What is the total runtime of the inside algoirthm?

- We can extract the marginal probability of $x_1 \ldots x_T$ as

$$p(x_1 \ldots x_T) = \sum_{\in \textsf{\textbf{GEN}}(x_1 \ldots x_T)} p(\quad) = \alpha(1,T,S)$$

- Base case?

Inside Algorithm

- The **inside algorithm** computes, bottom up, for all $1 \leq i \leq j \leq T$, for all $X \in N$,

$$\alpha(i,j,X) = \sum_{\;\in \textbf{GEN}(x_i \ldots x_j):\ \mathrm{root}(\;)=X} p(\;)$$

  We will see that computing each $\alpha(i,j,X)$ takes $O(T\,|R|)$ time.

  - What is the total runtime of the inside algoirthm?

- We can extract the marginal probability of $x_1 \ldots x_T$ as

$$p(x_1 \ldots x_T) = \sum_{\;\in \textbf{GEN}(x_1 \ldots x_T)} p(\;) = \alpha(1,T,S)$$

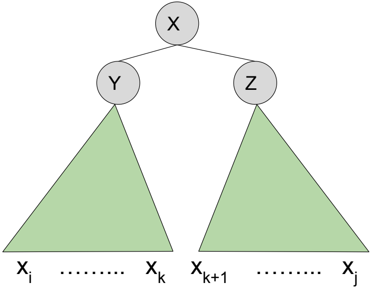
- Base case? $\alpha(i,i,X) = q(X \to x_i)$

Inside Algoirthm: Main Body

$\alpha(i,j,X) = \sum$ $p($  $)$

 $\in \textbf{GEN}(x_i \ldots x_j)$: root( $)=X$

$$= \sum_{\substack{i \leq k < j \\ X \to Y\ Z \in R}} q(X \to Y\ Z) \times \underbrace{\alpha(i,k,Y) \times \alpha(k+1,j,Z)}_{\text{combinatorial: all subtree combinations}}$$



CKY Parsing Algorithm

- The CKY algorithm computes, bottom up, for all $1 \leq i \leq j \leq T$, for all $X \in N$,

$$\pi(i,j,X) = \max_{\quad \in \textbf{GEN}(x_i \ldots x_j):\ \mathrm{root}(\quad)=X} p(\quad)$$

- Base: $\pi(i,j,X) = q(X \to x_i)$
- Main: $\pi(i,j,X) = \max_{i \leq k < j,\ X \to Y\ Z \in R} q(X \to Y\ Z) \times \pi(i,k,Y) \times \pi(k+1,j,Z)$
- The optimal probability and a backpointer for extracting the tree:

$$\pi(1,T,S) = \max_{\quad \in \textbf{GEN}(x_1 \ldots x_T)} p(\quad)$$

$$b(i,j,X) = \arg\max_{\substack{i \leq k < j \\ X \to Y\ Z \in R}} q(X \to Y\ Z) \times \pi(i,k,Y) \times \pi(k+1,j,Z)$$

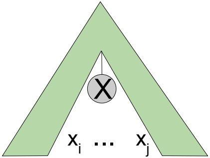
Computing Marginals Under PCFG

- Marginals

$$\mu(i, j, X) = \sum_{\in \textbf{GEN}(x_1 \ldots x_T):\ \mathrm{root}(\quad, i, j) = X} p(\quad)$$

- Need the **outside algorithm**

$$\beta(i, j, X) = \sum_{\in \textbf{OUT}(x_i \ldots x_j):\ \mathrm{foot}(\quad) = X} p(\quad)$$



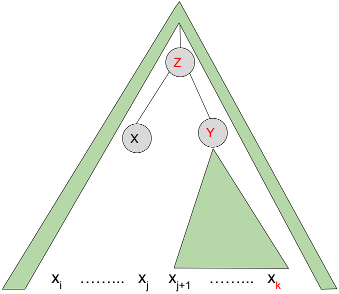
Outside Algorithm: Top-Down Marginalization

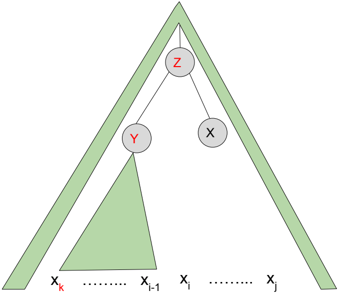
- **Base.** $\beta(1, T, S) = 1$ and $\beta(1, T, X) = 0$ for all $X \neq S$

- **Main.** For $l = T - 2 \ldots 1$, for $i = 1 \ldots T - l$ (set $j = i + l$), for $X \in N$,

$$\beta(i, j, X) = \sum_{\substack{j < k \leq T \\ Z \to X\ Y \in R}} \beta(i, k, Z) \times \alpha(j+1, k, Y) \times q(Z \to X\ Y) +$$

$$\sum_{\substack{1 \leq k < i \\ Z \to Y\ X \in R}} \beta(k, j, Z) \times \alpha(k, i-1, Y) \times q(Z \to Y\ X)$$





Max Marginal Parsing

- Inside-outside algorithm computes, for $1 \leq i \leq j \leq T$, for all $X \in N$,

$$\mu(i,j,X) = \sum_{\in \mathbf{GEN}(x_1 \ldots x_T):\ \mathrm{root}(\ ,i,j)=X} p(\ )$$

$$= \alpha(i,j,X) \times \beta(i,j,X)$$

- New parsing objective ($\neq$ CKY): find max marginal parse

$$^{*} = \arg\max_{\in \mathbf{GEN}(x_1 \ldots x_T)} \left(\sum_{(i,j,X) \in} \mu(i,j,X) \right)$$

- Labeled recall algorithm $O(T^3 \,|N|)$ (Goodman, 1996)

$$\gamma(i,j) = \max_X \mu(i,j,X) + \max_{i \leq k < j} \gamma(i,k) + \gamma(k+1,j)$$

Evaluating Parser Predictions

- Precision

$$p = \frac{\text{number of correctly predicted } (i, j, X)}{\text{number of predicted } (i, j, X)}$$

- Recall

$$r = \frac{\text{number of correctly predicted } (i, j, X)}{\text{number of ground-truth } (i, j, X)}$$

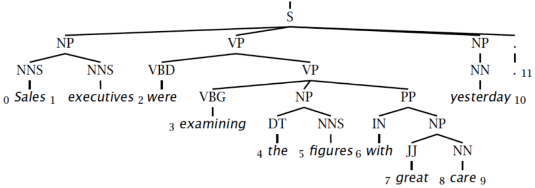
- Labeled F_1

$$F_1 = \frac{2 \times p \times r}{p + r}$$

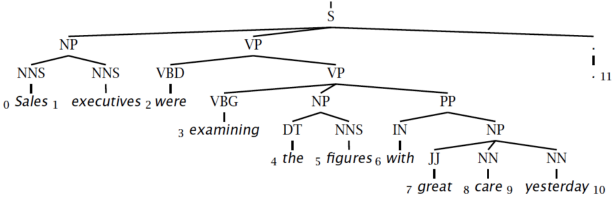
Can also consider unlabeled F_1

Example

Gold standard brackets: **S-(0:11)**, **NP-(0:2)**, VP-(2:9), VP-(3:9), **NP-(4:6)**, PP-(6-9), NP-(7,9), NP-(9:10)



Candidate brackets: **S-(0:11)**, **NP-(0:2)**, VP-(2:10), VP-(3:10), **NP-(4:6)**, PP-(6-10), NP-(7,10)



Precision 3/7 (42.9%), recall 3/8 (37.5%), labeled F_1 40