CRF example — 3/27/18
(CS 690N, UMass Amherst, Brendan O'Connor)

		finna	bless	us
y =	**[S]**	**V**	**V**	**V**

Tags: "**V**"erb and pr"**O**"noun (and **[S]**tart)
Let's use three feature templates:

Transition features: for example $f_{VV}(x,y)$ = number of V-V transitions in y

Word-tag observation features: for example $f_{V,dog}(x,y)$ = number of tokens that are word "dog" under a Verb tag

"ends with s"–tag features: $f_{V\text{-}s}(x,y)$ = number of tokens that end with -s and are tagged as Verb

(Global features have to be COUNTS: the reason why is further below.)
For 3 word vocabulary and 2 tag types, that's J=14 total features.
Assume we have fixed model weights θ and would like to score the goodness of the above tag sequence.

	fSV	fSO	fVV	fVO	fOV	fOO	fV,finna	fV,bless	fV,us	fO,finna	fO,bless	fO,us	fV,-s	fO,-s
Global feature vector f(x,y) =	1	0	2	0	0	0	1	1	1	0	0	0	2	0

	θSV	θSO	θVV	θVO	θOV	θOO	θV,finna	θV,bless	θV,us	θO,finna	θO,bless	θO,us	θV,-s	θO,-s
Model parameters θ =	-0.2	-0.8	+0.1	+0.5	+4.3	-0.3	-1.2	-0.1	+0.1	+5.3	-4.1	-0.3	+1.1	+2.2

Goodness score G(y) $= \theta' f(x,y) = \sum_{j=1}^{J} \theta_j f_j(x,y)$

$= -0.2 + 0 + 0.2 + 0 + 0 + 0 \quad -1.2 + 0 + 0.1 + 5.3 + 0 + 0 + 2.2 + 0$

Global feature vector is from the sum of local feature vectors

$$f(x,y) = \sum_t f_t(y_{t-1}, y_t, x_t)$$

$f_t(y_{t-1}, y_t, x_t) =$ local feature vector including the transition between these two tags, and the observation of word at position t.

The local features are, for example:

f_{VV}(yprev, ycur, curword) = {1 if yprev=V and ycur=V, else 0}

$f_{V,dog}$(yprev, ycur, curword) = {1 if ycur=V and curword="dog", else 0}

$f_{V,\text{-}s}$(yprev, ycur, curword) = {1 if ycur=V and curword ends in "s", else 0}

And so on, repeated for different tags and words.

Example

	fSV	fSO	fVV	fVO	fOV	fOO	fV,finna	fV,bless	fV,us	fO,finna	fO,bless	fO,us	fV,-s	fO,-s
	………trans. feats………						……………obs. feats……………							
f(START, V, finna)	1	0	0	0	0	0	1	0	0	0	0	0	0	0
+ f(V, V, bless)	0	0	1	0	0	0	0	1	0	0	0	0	1	0
+ f(V, V, us)	0	0	1	0	0	0	0	0	1	0	0	0	1	0
= f(x=finna bless us, y=V V V) =	1	0	2	0	0	0	1	1	1	0	0	0	2	0

Local feature decomposition implies that the scoring function decomposes, too.

$$G(y) = \theta' f(x,y) = \theta' \sum_t f_t(y_{t-1}, y_t, x_t) = \sum_t \theta' f_t(y_{t-1}, y_t, x_t)$$

= θ'f(START, V, finna) + θ'f(V, V, bless) + θ'f(V, V, us)

= dotprod (

-0.2	-0.8	+0.1	+0.5	+4.3	-0.3	-1.2	-0.1	+0.1	+5.3	-4.1	-0.3	+1.1	+2.2
1	0	0	0	0	0	1	0	0	0	0	0	0	0

)

\+ dotprod (

-0.2	-0.8	+0.1	+0.5	+4.3	-0.3	-1.2	-0.1	+0.1	+5.3	-4.1	-0.3	+1.1	+2.2
0	0	1	0	0	0	0	1	0	0	0	0	1	0

)

\+ dotprod (

-0.2	-0.8	+0.1	+0.5	+4.3	-0.3	-1.2	-0.1	+0.1	+5.3	-4.1	-0.3	+1.1	+2.2
0	0	1	0	0	0	0	0	1	0	0	0	1	0

)

On the whiteboard we talked about compiling these into factor scoring functions.

ϕ_t(yprev, ycur) is a matrix of "pairwise goodness scores" (a.k.a. *log-potentials*) that summarize the model's soft constraints between the tags at t-1 and t. There is one such matrix for each position. The collection of all these matrices is the input for forward-backward or Viterbi. Note they include both transition *and* emission information. Definition:

$$\phi_t(y_{t-1}, y_t) \equiv \theta' f_t(y_{t-1}, y_t, x_t)$$

For an HMM, this is:

$$\phi_t(y_{t-1}, y_t) = \log p(y_t \mid y_{t-1}) + \log p(w_t \mid y_t)$$

We can equivalently write the transition/emission probability lookups as θ'f(.) dot-products, where A and B range over all tags:

$$\phi_t(y_{t-1}, y_t) = \left(\sum_{A,B} \theta_{A,B}\, 1\{y_{t-1} = A, y_t = B\} \right) + \left(\sum_A \theta_{A,w_t}\, 1\{y_t = A\} \right)$$