СКРЫТЫЕ МАРКОВСКИЕ МОДЕЛИ II

РАСШИРЕНИЯ И РАСПОЗНАВАНИЕ РЕЧИ

Сергей Николенко

НИУ ВШЭ — Санкт-Петербург
26 мая 2017 г.

Random facts:

- 26 мая 1908 в Персии Уильям д'Арси начал добывать первую на Ближнем Востоке нефть
- 26 мая 1931 г. в Харбине была создана крупнейшая организация в среде русской эмиграции — Российская фашистская партия

Специальные виды марковских моделей

Непрерывные плотности наблюдаемых

- У нас были дискретные наблюдаемые с вероятностями $B = (b_j(k))$.
- Но в реальной жизни всё сложнее: зачастую мы наблюдаем непрерывные сигналы, а не дискретные величины, и дискретизовать их или плохо, или неудобно.
- При этом саму цепь можно оставить дискретной, т.е. перейти к непрерывным $b_j(D)$.

Специальный вид плотности

- Не для всех плотностей найдены алгоритмы пересчёта (обобщения алгоритма Баума–Уэлча).
- Наиболее общий результат верен, когда $b_j(D)$ можно представить в виде

$$b_j(D) = \sum_{m=1}^{M} c_{jm} \mathcal{P}(D, \mu_{jm}, \sigma_{jm}),$$

  где c_{jm} — коэффициенты смеси ($\sum_m c_{jm} = 1$), а $\mathcal{P}$ — выпуклое распределение со средним μ и вариацией σ (гауссиан подойдёт).
- К счастью, такой конструкцией можно приблизить любое непрерывное распределение, поэтому это можно широко применять.

ВСПОМОГАТЕЛЬНЫЕ ПЕРЕМЕННЫЕ

- $\gamma_t(j, m)$ — вероятность быть в состоянии j во время t, причём за D отвечает m–й компонент смеси.
- Формально говоря,

$$\gamma_t(j, m) = \left[\frac{\alpha_t(j)\beta_t(j)}{\sum_{j=1}^{N} \alpha_t(j)\beta_t(j)} \right] \left[\frac{c_{jm}\mathcal{P}(d_t, \mu_{jm}, \sigma_{jm})}{\sum_{m=1}^{M} c_{jm}\mathcal{P}(d_t, \mu_{jm}, \sigma_{jm})} \right].$$

- Если $M = 1$, то это уже известные нам $\gamma_t(j)$.

Алгоритм для этого случая

- Нужно научиться пересчитывать $b_j(D)$, т.е. пересчитывать c_{jm}, μ_{jm} и σ_{jm}.
- Это делается так:

$$\bar{c}_{jm} = \frac{\sum_{t=1}^{T} \gamma_t(j,m)}{\sum_{t=1}^{T} \sum_{m=1}^{M} \gamma_t(j,m)},$$

$$\bar{\mu}_{jm} = \frac{\sum_{t=1}^{T} \gamma_t(j,m) \cdot d_t}{\sum_{t=1}^{T} \gamma_t(j,m)},$$

$$\bar{\sigma}_{jm} = \frac{\sum_{t=1}^{T} \gamma_t(j,m) \cdot (d_t - \mu_{jm})(d_t - \mu_{jm})^t}{\sum_{t=1}^{T} \gamma_t(j,m)}.$$

ПРОБЛЕМА

- Как моделировать продолжительность нахождения в том или ином состоянии?
- В дискретном случае вероятность пробыть в состоянии i d шагов:

$$p_i(d) = a_{ii}^{d-1}(1 - a_{ii}).$$

- Однако для большинства физических сигналов такое экспоненциальное распределение не соответствует действительности. Мы бы хотели явно задавать плотность пребывания в данном состоянии.
- Т.е. вместо коэффициентов перехода в себя a_{ii} — явное задание распределения $p_i(d)$.

Вспомогательные переменные

- Введём переменные

$$\alpha_t(i) = p(d_1 \dots d_t, x_i \text{ заканчивается во время } t|\lambda).$$

- Всего за первые t шагов посещено r состояний $q_1 \dots q_r$, и мы там оставались $d_1, \dots, d_r$. Т.е. ограничения:

$$q_r = x_i, \qquad \sum_{s=1}^{r} d_s = t.$$

ВЫЧИСЛЕНИЕ $\alpha_t(i)$

- Тогда получается

$$\alpha_t(i) = \sum_q \sum_d \pi_{q_1} p_{q_1}(d_1) p(d_1 d_2 \dots d_{d_1} | q_1)$$
$$a_{q_1 q_2} p_{q_2}(d_2) p(d_{d_1+1} \dots d_{d_1+d_2} | q_2) \dots$$
$$\dots a_{q_{r-1} q_r} p_{q_r}(d_r) p(d_{d_1+\dots+d_{r-1}+1} \dots d_t | q_r).$$

ВЫЧИСЛЕНИЕ $\alpha_t(i)$

- По индукции

$$\alpha_t(j) = \sum_{i=1}^{n} \sum_{d=1}^{D} \alpha_{t-d}(j) a_{ij} p_j(d) \prod_{s=t-d+1}^{t} b_j(d_s),$$

  где D — максимальная остановка в любом из состояний.

- Тогда, как и раньше,

$$p(d|\lambda) = \sum_{i=1}^{n} \alpha_T(i).$$

Вспомогательные переменные

- Для пересчёта потребуются ещё три переменные:

$$\alpha_t^*(i) = p(d_1 \dots d_t, x_i \text{ начинается во время } t+1|\lambda),$$

$$\beta_t(i) = p(d_{t+1} \dots d_T|x_i \text{ заканчивается во время } t, \lambda),$$

$$\beta_t^*(i) = p(d_{t+1} \dots d_T|x_i \text{ начинается во время } t+1, \lambda).$$

Вспомогательные переменные

- Соотношения между ними:

$$\alpha_t^*(j) = \sum_{i=1}^{n} \alpha_t(i) a_{ij},$$

$$\alpha_t(i) = \sum_{d=1}^{D} \alpha_{t-d}^*(i) p_i(d) \prod_{s=t-d+1}^{t} b_i(d_s),$$

$$\beta_t(i) = \sum_{j=1}^{n} a_{ij} \beta_t^*(j),$$

$$\beta_t^*(i) = \sum_{d=1}^{D} \beta_{t+d}(i) p_i(d) \prod_{s=t+1}^{t+d} b_i(d_s).$$

ФОРМУЛЫ ПЕРЕСЧЁТА

- Приведём формулы пересчёта.
- π_i — просто вероятность того, что x_i был первым состоянием:

$$\hat{\pi}_i = \frac{\pi_i \beta_0^*(i)}{p(d|\lambda)}.$$

- a_{ij} — та же формула, что обычно, только вместе с α есть ещё и β, которая говорит, что новое состояние начинается на следующем шаге:

$$\hat{a}_{ij} = \frac{\sum_{t=1}^{T} \alpha_t(i) a_{ij} \beta_t^*(j)}{\sum_{k=1}^{n} \sum_{t=1}^{T} \alpha_t(i) a_{ik} \beta_t^*(k)}.$$

ФОРМУЛЫ ПЕРЕСЧЁТА

- $b_i(k)$ — отношение ожидания количества событий $d_t = v_k$ в состоянии x_i к ожиданию количества любого v_j в состоянии x_i:

$$\hat{b}_i(k) = \frac{\sum_{t=1, d_t=v_k}^{T} \left(\sum_{\tau<t} \alpha^*_\tau(i)\beta^*_\tau(i) - \sum_{\tau<t} \alpha_\tau(i)\beta_\tau(i) \right)}{\sum_{k=1}^{m} \sum_{t=1, d_t=v_k}^{T} \left(\sum_{\tau<t} \alpha^*_\tau(i)\beta^*_\tau(i) - \sum_{\tau<t} \alpha_\tau(i)\beta_\tau(i) \right)}.$$

- $p_i(d)$ — отношение ожидания количества раз, которые x_i случилось с продолжительностью d, к количеству раз, которые x_i вообще случалось:

$$\hat{p}_i(d) = \frac{\sum_{t=1}^{T} \alpha^*_t(i) p_i(d) \beta_{t+d}(i) \prod_{s=t+1}^{t+d} b_i(d_s)}{\sum_{d=1}^{D} \sum_{t=1}^{T} \alpha^*_t(i) p_i(d) \beta_{t+d}(i) \prod_{s=t+1}^{t+d} b_i(d_s)}.$$

ЗА И ПРОТИВ

- Такой подход очень полезен, когда $p_i(d)$ далеко от экспоненциального.
- Однако он сильно увеличивает вычислительную сложность (в D^2 раз).
- И, кроме того, становится гораздо больше параметров, т.е. нужно, вообще говоря, больше данных, чтобы эти параметры надёжно оценить.

ПАРАМЕТРИЧЕСКАЯ ПРОДОЛЖИТЕЛЬНОСТЬ СОСТОЯНИЯ

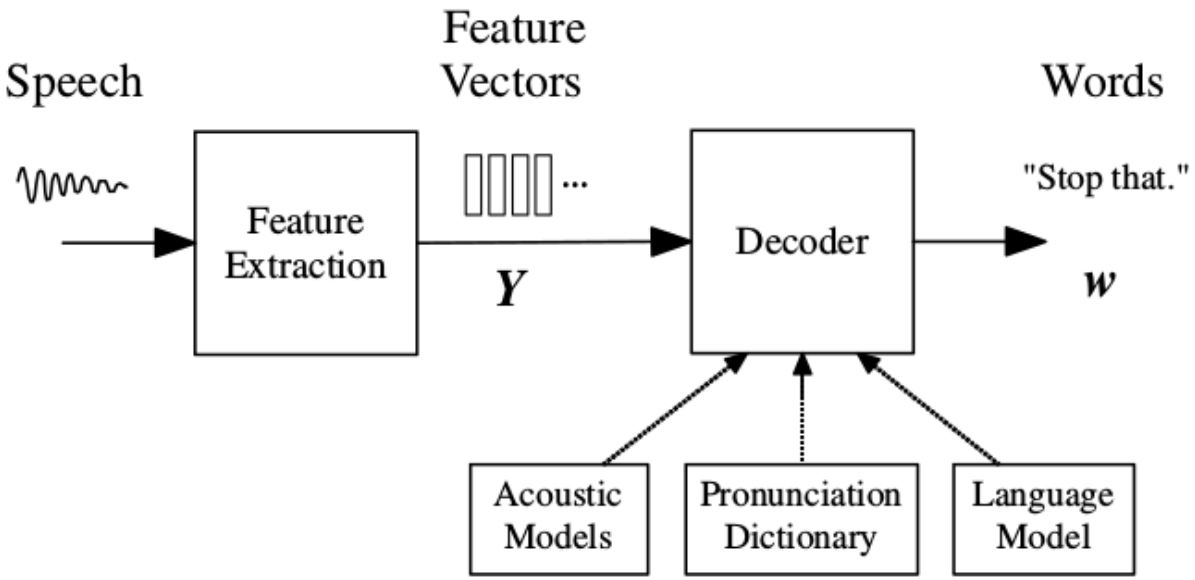
- Чтобы уменьшить количество параметров, можно иногда считать, что $p_i(d)$ — классическое распределение с не слишком большим количеством параметров.
- Например, $p_i(d)$ может быть равномерным, или нормальным ($p_i(d) = \mathcal{N}(d, \mu_i, \sigma_i^2)$), или гамма–распределением:

$$p_i(d) = \frac{\eta_i^{\nu_i} d^{\nu_i - 1} e^{-\eta_i d}}{\Gamma(\nu_i)}.$$

HMM для распознавания речи

ОБЩАЯ СТРУКТУРА

- HMM – классический подход к распознаванию речи.
- Сейчас, правда, они уже в основном заменены глубокими нейронными сетями; но всё равно полезно проследить, как их можно применить.



ОТ СЛОВ К ФОНЕМАМ

- Признаки как-то выделились, а потом слово w делится на фонемы $\mathbf{q} = q_1 \dots q_{|w|}$, и правдоподобие наблюдаемых признаков

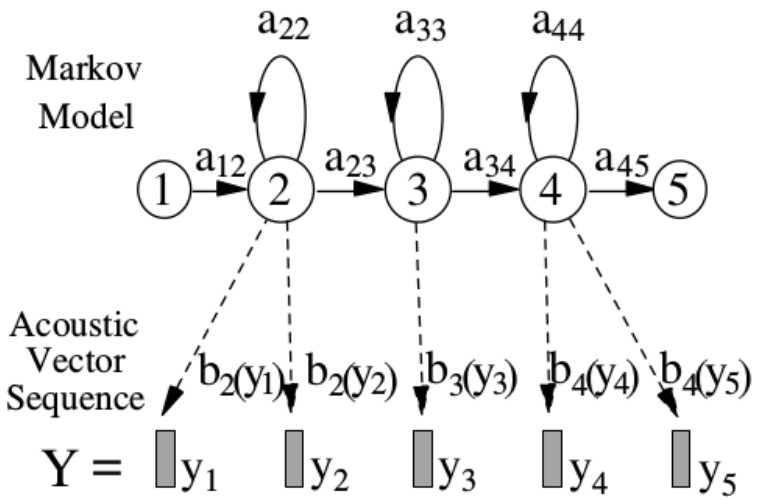
$$p(\mathbf{y} \mid \mathbf{w}) = \sum_{\mathbf{q}} p(\mathbf{y} \mid \mathbf{q}) p(\mathbf{q} \mid \mathbf{w}),$$

  сумма по возможным произношениям (маленькая сумма), а $\mathbf{w}$ – это все слова $w_1 \dots w_L$:

$$p(\mathbf{q} \mid \mathbf{w}) = \prod_{l=1}^{L} p(\mathbf{q}^{(w_l)} \mid w_l).$$

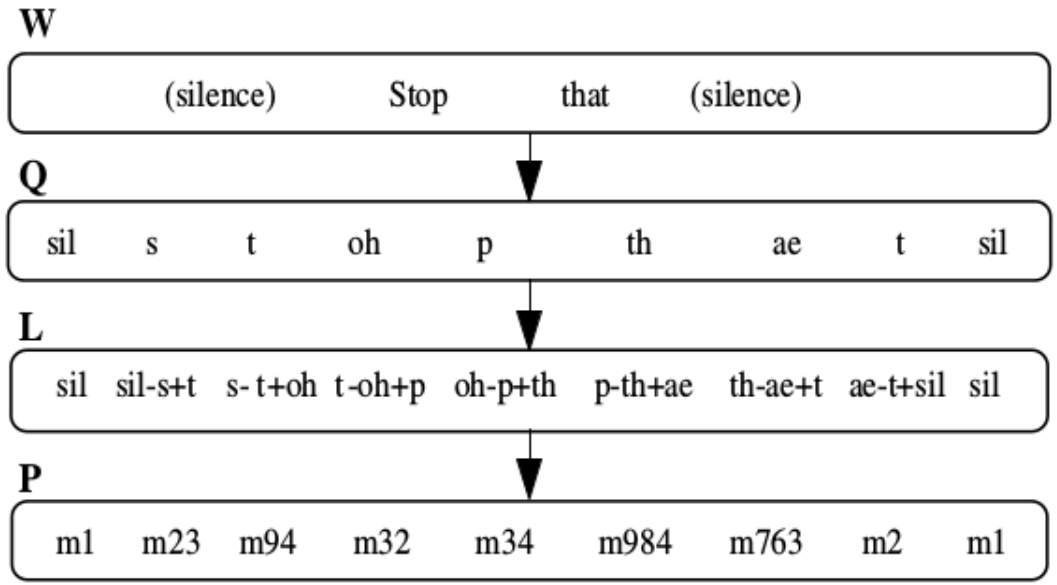
АКУСТИЧЕСКАЯ МОДЕЛЬ

- Каждая фонема – это HMM с непрерывными наблюдаемыми $b_j(\mathbf{y}) = \mathcal{N}(\mathbf{y}; \mu^{(j)}, \Sigma^{(j)})$.
- На этом месте уже можно обучать просто всё сразу.



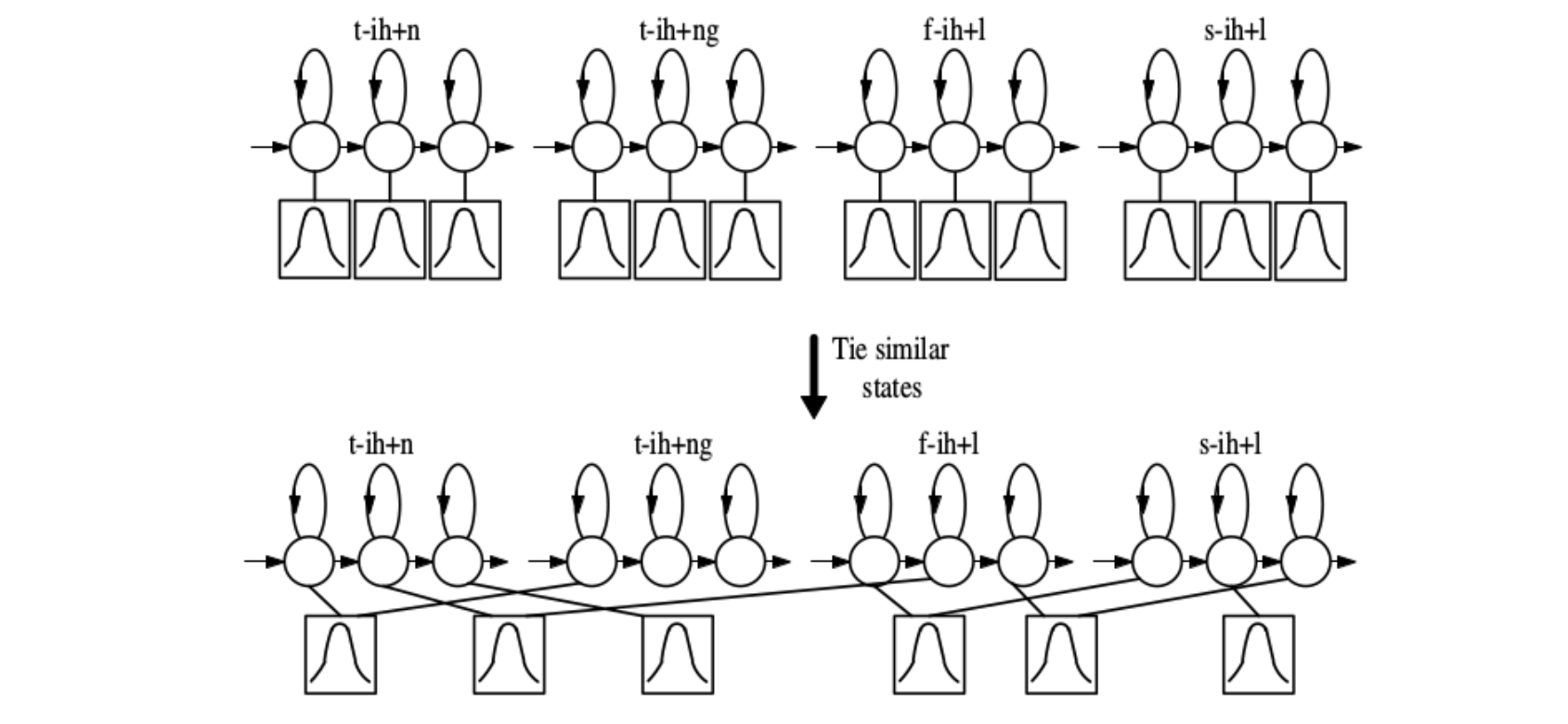
АКУСТИЧЕСКАЯ МОДЕЛЬ

- Однако фонемы очень по-разному звучат в зависимости от контекста.
- Можно перейти к трифонам.



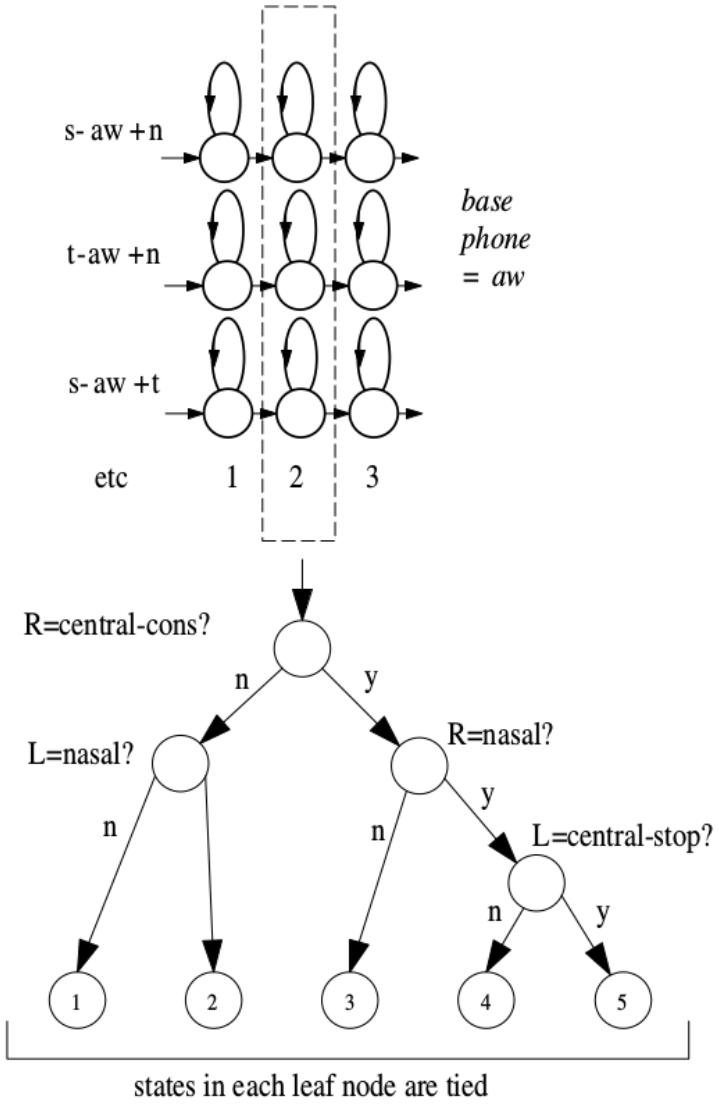
АКУСТИЧЕСКАЯ МОДЕЛЬ

- Но их будет целых N^3, и лучше объединить похожие и связать их параметры:



АКУСТИЧЕСКАЯ МОДЕЛЬ

- Это можно сделать просто силой мысли:



Языковая модель

- Но это только начало. Ещё нужна *языковая модель* – мы о многом просто догадываемся.
- То есть нужно априорное распределение

$$p(\mathbf{w}) = \prod_{l=1}^{L} p(w_l \mid w_1 \dots w_{l-1}).$$

- Классический подход – n-граммы для $n = 2..4$:

$$p(\mathbf{w}) = \prod_{l=1}^{L} p(w_l \mid w_{l-1} \dots w_{l-n+1}).$$

- Качество языковых моделей сравнивают в терминах их *перплексии* (perplexity)

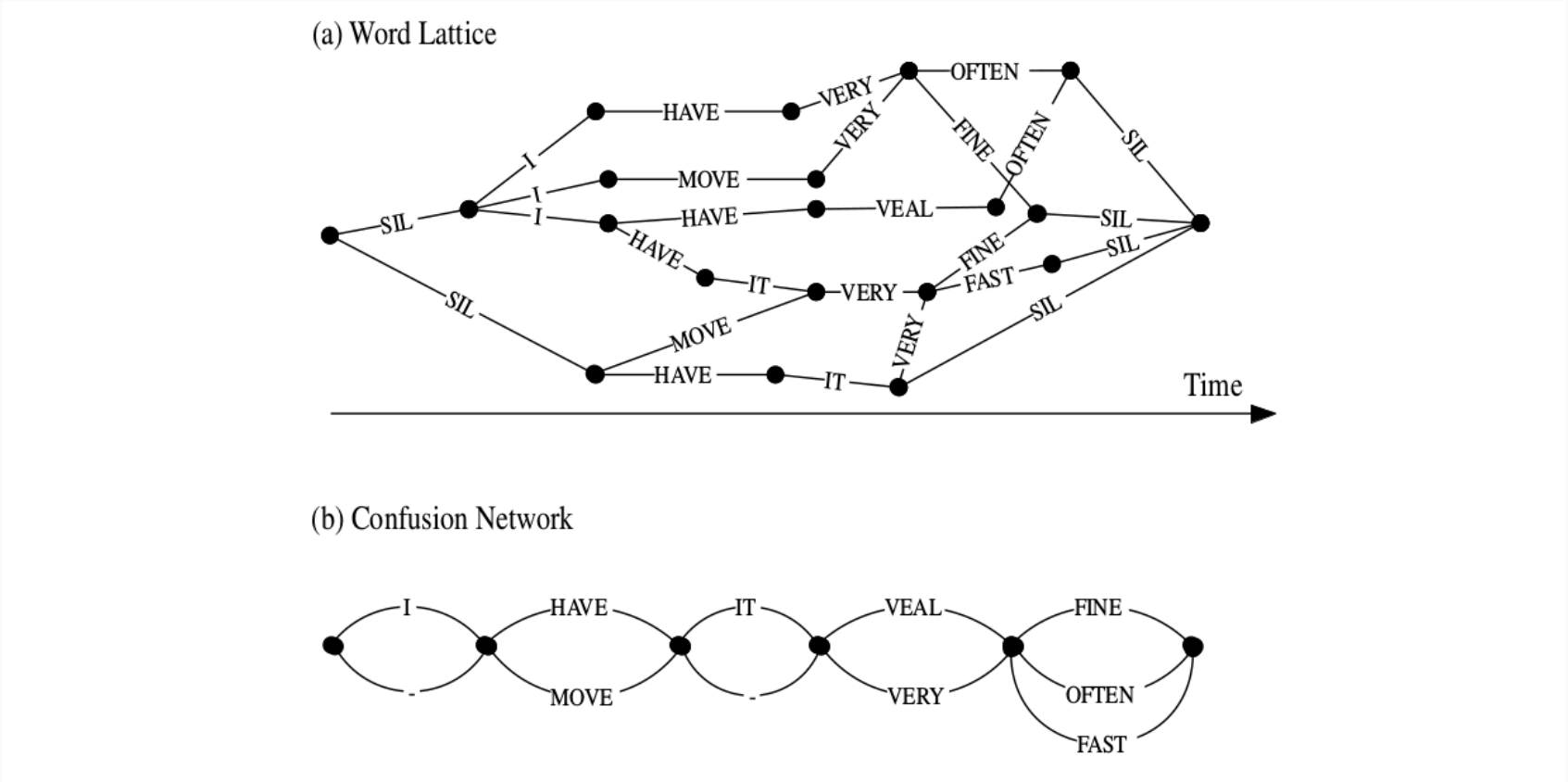
$$H = -\lim_{L\to\infty} \frac{1}{K} \log p(w_1, \dots, w_L).$$

РЕЗУЛЬТАТЫ

- О современных языковых моделях мы обязательно поговорим потом...
- А пока декодер идёт и алгоритмом Витерби всё решает.
- Но что именно решает?

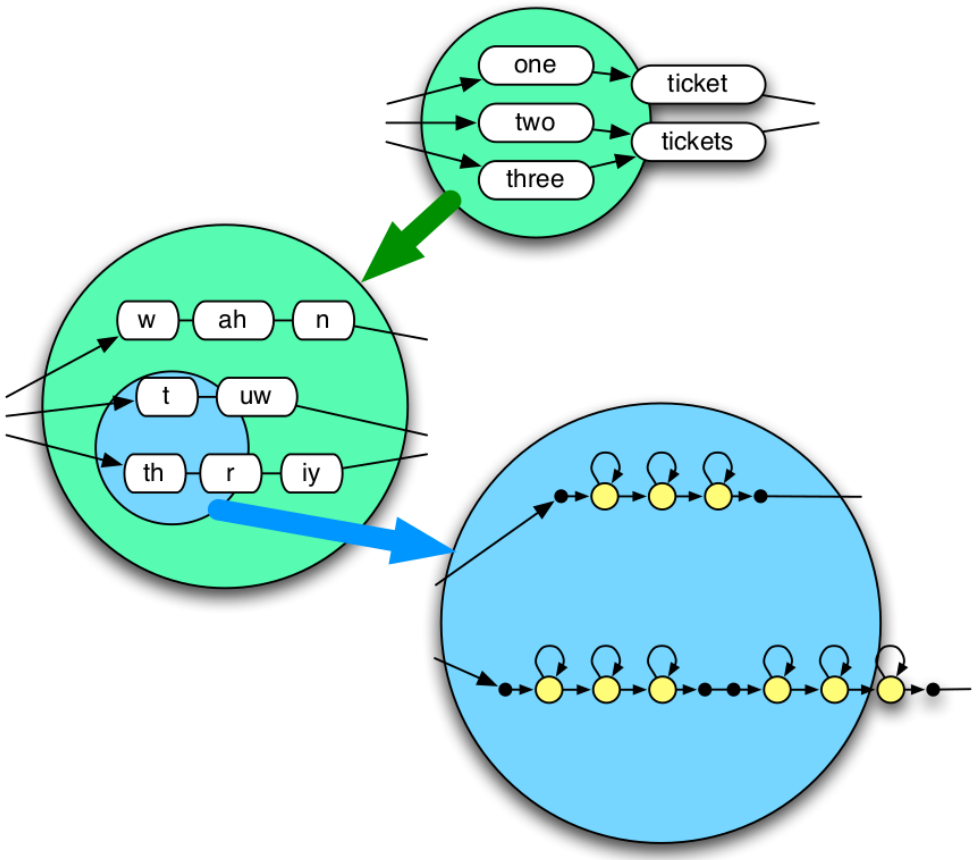
РЕЗУЛЬТАТЫ

- Возможности удобно представлять как *решётку слов* (word lattice) или *confusion network*.



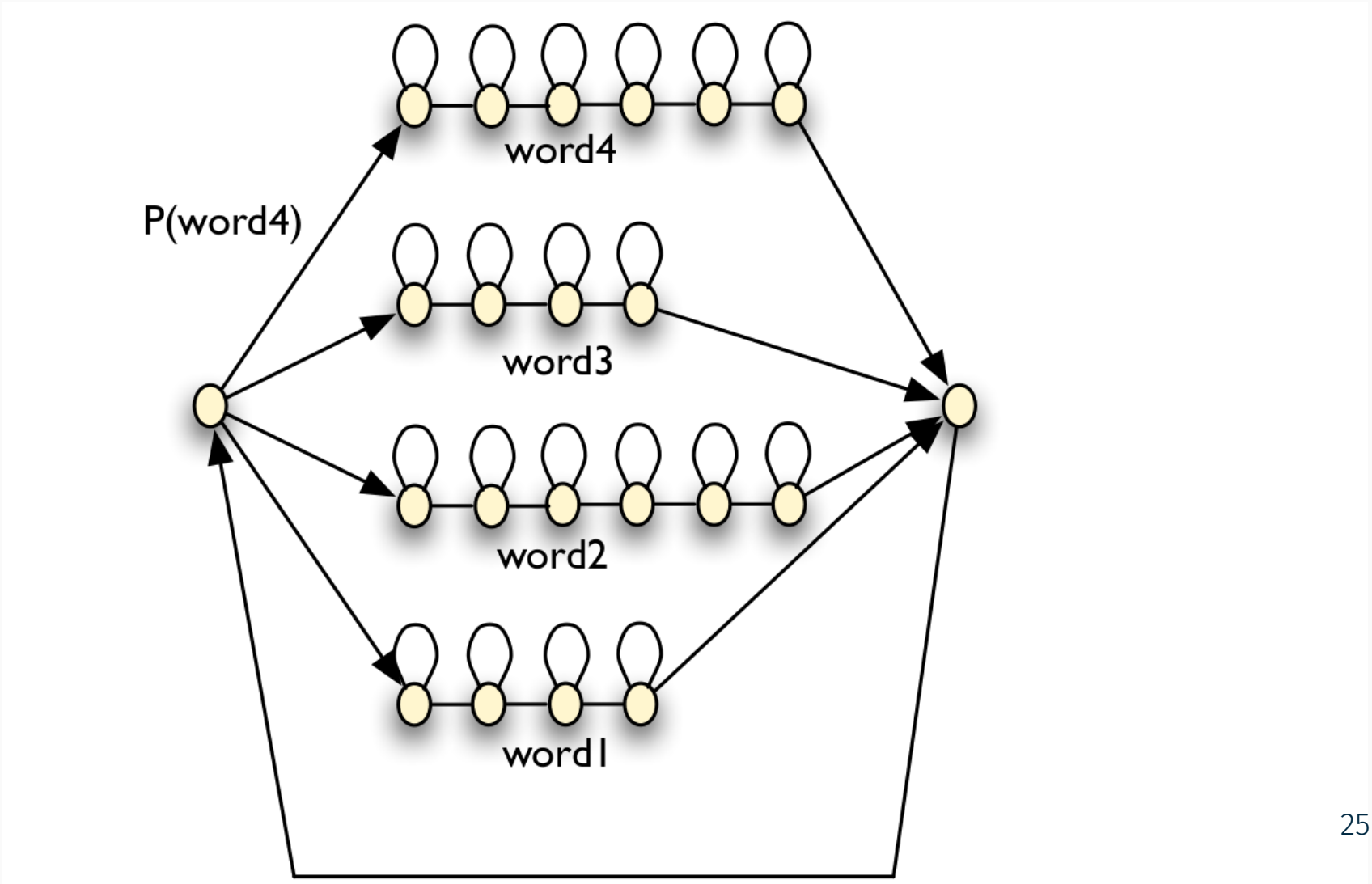
РЕЗУЛЬТАТЫ

- Сеть слов превращается в сеть фонем, потом в сеть состояний HMM.



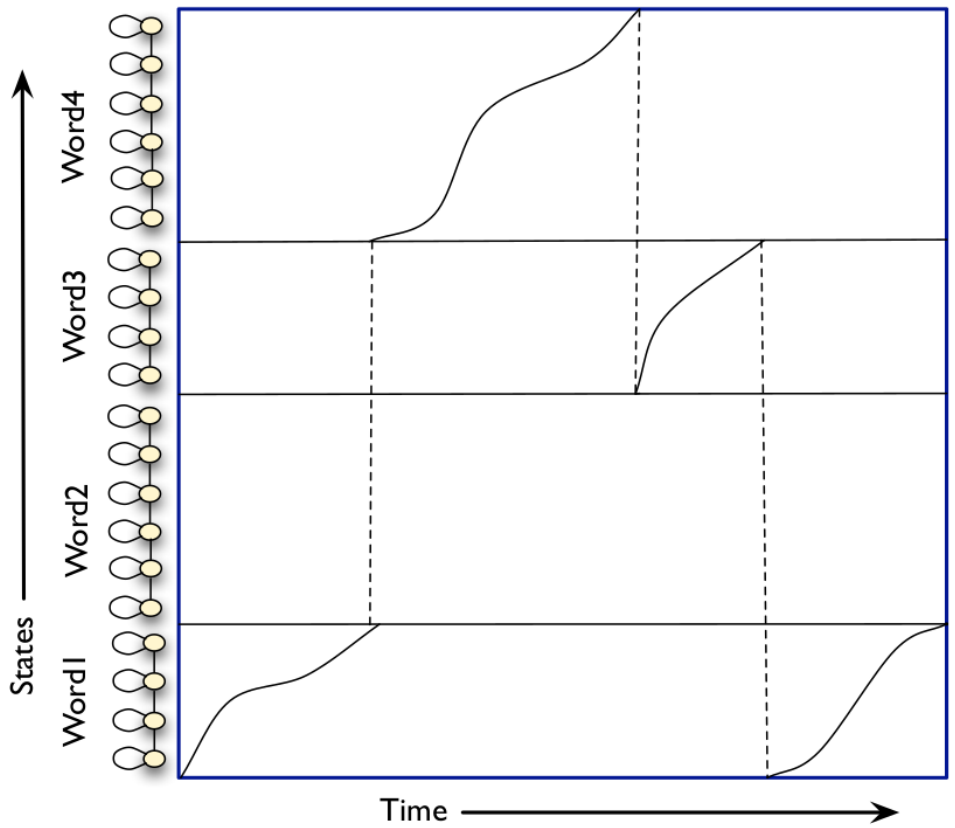
РЕЗУЛЬТАТЫ

- И можно распознавать связанные друг с другом слова тоже алгоритмом Витерби.



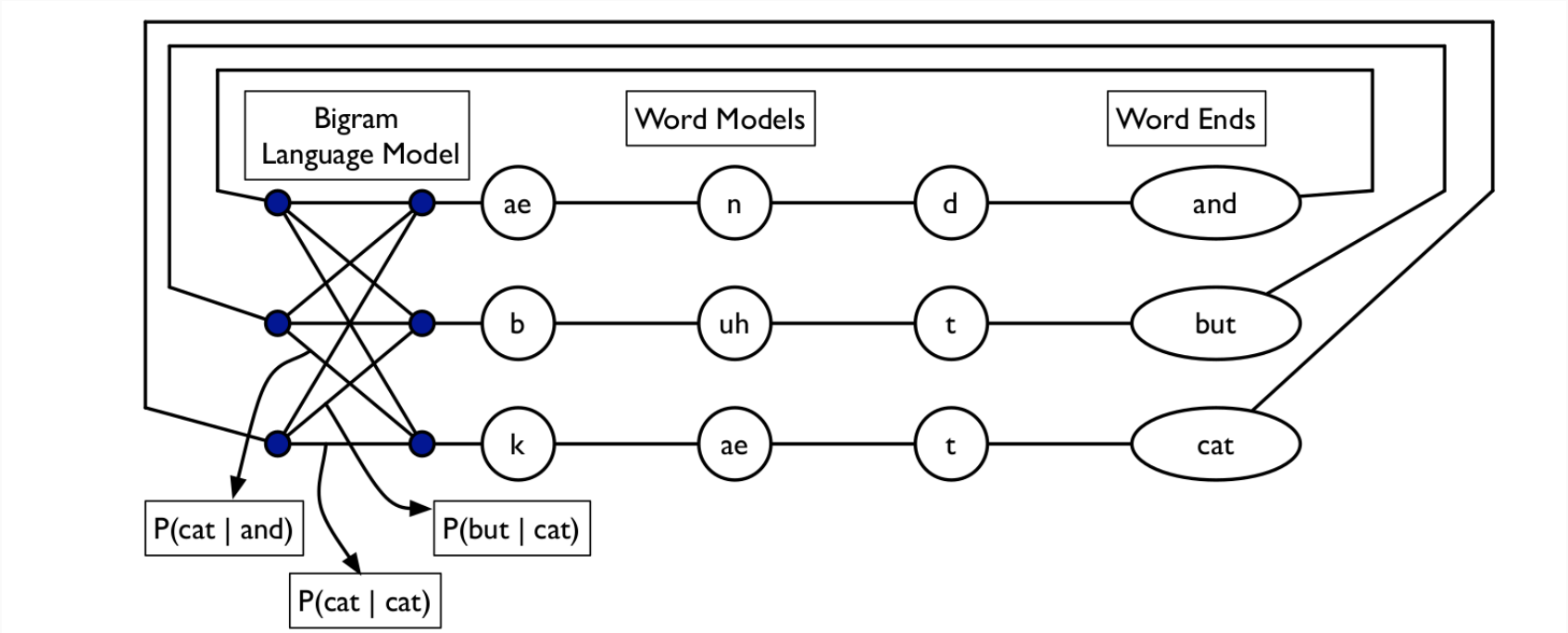
РЕЗУЛЬТАТЫ

- Всё это накладывается на собственно аудиозапись, получается последовательность во времени.



РЕЗУЛЬТАТЫ

- А языковая модель – это просто дополнительные множители (слагаемые в $\log$), априорные вероятности.

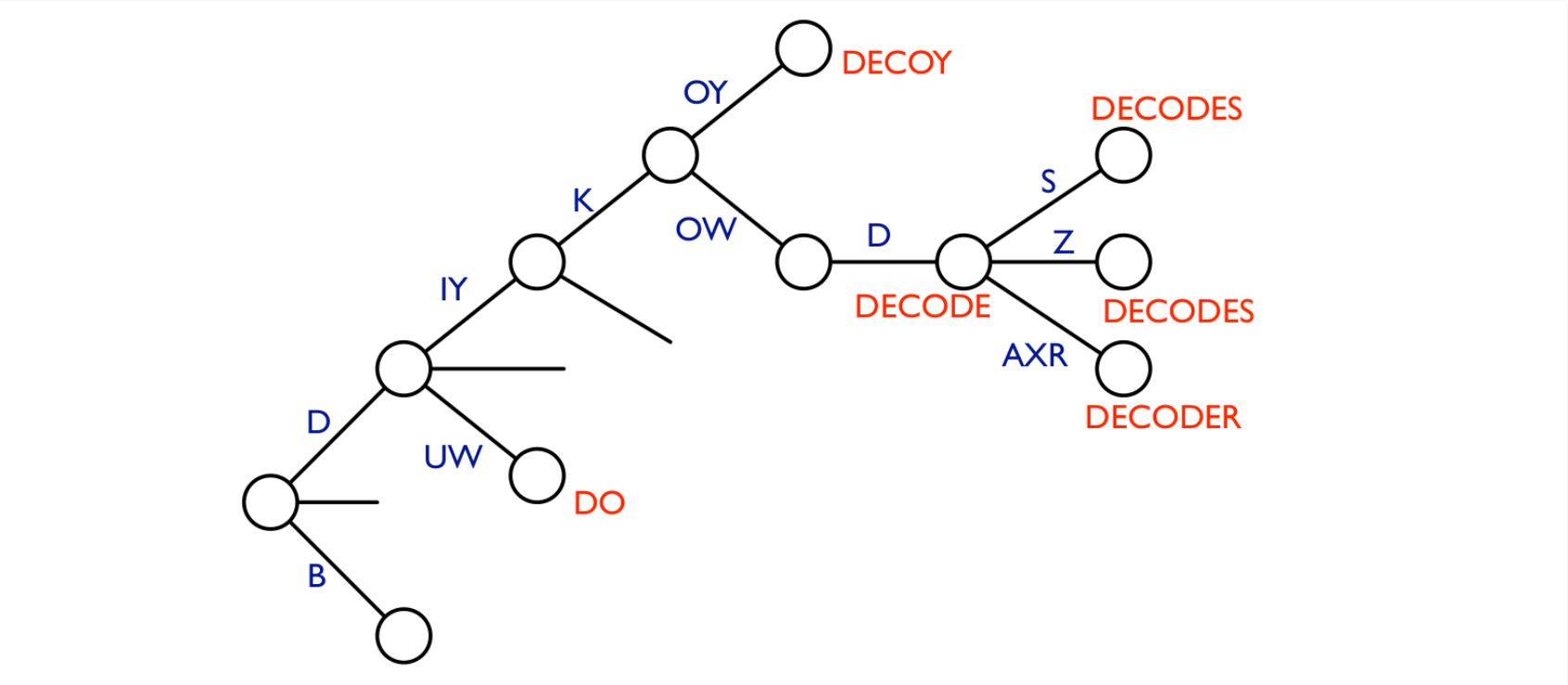


РЕЗУЛЬТАТЫ

- Декодирование по всей сети всего языка нереально.
- Обычно декодер генерирует и поддерживает только лучшие гипотезы; это называется *beam search*.
- Т.е. мы после каждого шага (обычно на границах слов) делаем pruning и либо выкидываем гипотезы, которые сильно хуже текущего лучшего варианта, либо просто поддерживаем N лучших (типа 1000).

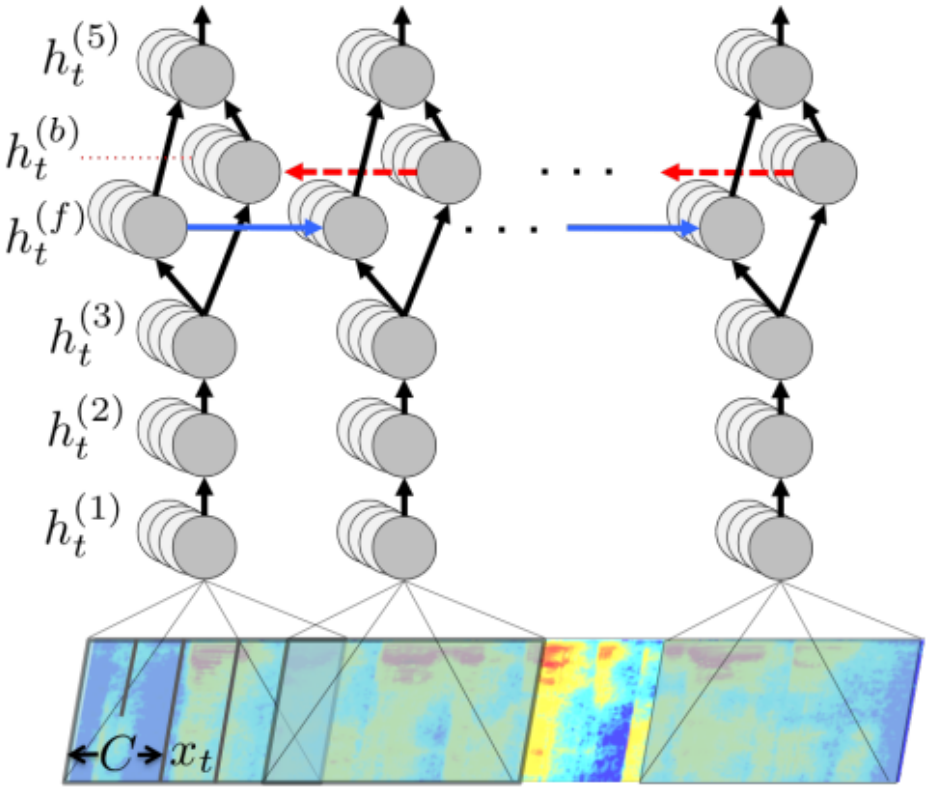
РЕЗУЛЬТАТЫ

- А HMM для отдельных слов (нам же нужно по HMM на каждое слово) тоже можно организовать в дерево (префиксное).



END-TO-END

- Впрочем, сейчас уже всё по-другому.
- End-to-end speech recognition.
- Но об этом – потом.



СПАСИБО!

Спасибо за внимание!