0. Intro

Hypergraph View: Computation graph = labeled acyclic **hypergraph**. Edges can have multiple sources/targets. **Complexity**: same time as f; space higher (store intermediates) vec-vec: $O(d)$; mat-vec: $O(nm)$; mat-mat: $O(nm\ell)$

NLL ∇ = 0: $\sum_{i=1}^n f(x_i,y_i) = \sum_{i=1}^n \mathbb{E}_{y|x_i,\theta}[f(x_i,y)]$ Observed features = Expected features **Hessian**: $H = \sum_i \mathrm{Cov}_{y|x_i,\theta}[f(x_i,y)]$ (PSD!)

DAG Properties: Topological order 唯一确定; DP 子问题独立拆分可行; Gradient 反向传播良定义(no cycles) **Hypergraph**: 函数式计算自然表示, multi-inputs→one output

1. Backpropagation

Chain: $\frac{d}{dx}[f(g(x))] = f'(g(x))g'(x)$ **Jacobian**: $f: \mathbb{R}^n\to\mathbb{R}^m$, $\frac{dy}{dx} = \left[\frac{dy}{dx_1},\dots,\frac{dy}{dx_n}\right] \in \mathbb{R}^{m\times n}$ **Multivar**: $\frac{dy_i}{dx_j} = \sum_{k=1}^m \frac{dy_i}{dz_k}\frac{dz_k}{dx_j}$

Bauer Path: $\frac{dy_i}{dx_j} = \sum_{p\in\mathcal{P}(j,i)}\prod_{(k,l)\in p}\frac{dz_l}{dz_k}$ $\mathcal{P}(j,i)$=all paths $j\to i$; worst $O(m^n)$, m平均出度, n路径长度

Forward vs Reverse: **Forward**: expand $\frac{d}{dx}$ recursively, same flow as fwd **Reverse**: 2 passes—fwd compute vals, bwd compute grads **Complexity**: same time as f; higher space (store intermediates)

Primitives: **Sum**: $\frac{d(a+b)}{da} = 1$; **Prod**: $\frac{d(ab)}{da} = b$

2. Log-Linear Models

Prob Basics: **Bayes**: $p(y|x) = \frac{p(x|y)p(y)}{\int p(x|y)p(y)dy}$ Posterior $\propto$ Prior $\times$ Likelihood **Marginal**: $p(x) = \sum_y p(x,y)$ **Expectation**: $\mathbb{E}[f(x)] = \sum_x f(x)p(x)$

Log-Linear Model: $p(y|x,\theta) = \frac{\exp(\theta\cdot f(x,y))}{Z(\theta)}$ $Z(\theta) = \sum_{y'\in Y}\exp(\theta\cdot f(x,y'))$ $\log p(y|x,\theta) = \theta\cdot f(x,y) - \log Z$ (linear in log space!) **Discrete MLE**: $p(y|x) = \frac{\text{count}(x,y)}{\text{count}(x)}$ (sparse 问题)

MLE ∇: $\theta_{\text{MLE}} = \arg\min_\theta -\sum_{n=1}^N \log p(y_n|x_n,\theta)$ 观测特征 count = 期望特征 count → **Expectation Matching** $\frac{d\mathcal{L}}{d\theta_k} = -\sum_n f_k(x_n,y_n) + \sum_n\sum_{y'} p(y'|x_n;\theta) f_k(x_n,y')$

MAP & Ridge: $\hat\theta_{\text{MAP}} = \arg\min[-\log p(\theta) - \log p(D|\theta)]$ Gaussian prior $\mathcal{N}(0,\sigma_p^2 I)$ → L2: $\frac{\lambda}{2}\|\theta\|^2$ Laplace prior → L1 regularization

Softmax: $\text{sftm}(h,y,T) = \frac{\exp(h_y/T)}{\sum_{y'}\exp(h_{y'}/T)}$ $T\to 0$: argmax; $T\to\infty$: uniform $\log\text{sftm} = h_y - \log\sum_{y'}\exp(h_{y'})$ (logsumexp)

MLP Architecture: **Problem**: Log-linear needs linearly separable data **Solution**: Learn non-linear feature fn $h_k = \sigma_k(W_k^\top h_{k-1})$, $h_1 = \sigma_1(W_1^\top e(x))$ Output: $\text{sftm}(\theta^\top h_n)$

Activations: $\sigma(x) = \frac{1}{1+\exp(-x)}$, $\nabla\sigma = \sigma(1-\sigma)$ **tanh**: $\frac{1-e^{-2x}}{1+e^{-2x}}$, $\nabla = 1-\tanh^2$ Sigmoid/tanh vanishing gradient→ use ReLU **Backprop**: $\frac{\partial\ell}{\partial W_k} = \frac{\partial\ell}{\partial y}\frac{\partial y}{\partial h_n}\left(\prod_{m=k+1}^n \sigma_m' W_m\right)\sigma_k' h_{k-1}$

Exp Family & MaxEnt: $p(x|\theta) = \frac{1}{Z(\theta)}h(x)\exp(\theta\cdot\varphi(x))$ **Max Entropy**: $H(p) = -\sum_x p(x)\log p(x)$ 选最大熵分布=最少假设=Laplace 原则 优势: Conjugate priors; Sufficient stats; Convex log-partition → unique MLE

3. Language Models

Structured Prediction: **Kleene V^***: infinite set of finite-length strings from V **Language Model**: weighted prefix tree, each sentence=unique path $p(y) = \frac{1}{Z}\prod_{t=1}^{|y|}\text{weight}_{y_{\le t}}$

Local Normalization: $Z=1$ when children edges sum to 1 at each node **Consistency**: $p(\text{EOS}|y_{<t},V^*) > \varepsilon > 0$ $p(|y|=\infty) \le \lim_{t\to\infty}(1-\varepsilon)^t = 0$ (tight)

N-gram Model: $p(y_t|y_{<t}) = p(y_t|y_{t-1},\dots,y_{t-n+1})$ **Markov**: $P(t_i|t_{1:i-1}) = P(t_i|t_{i-1})$ (1st order) $= \frac{\exp(w_{y_t}\cdot h_t)}{\sum_{y'\in V}\exp(w_{y'}\cdot h_t)}$, $h_t\in\mathbb{R}^d$ **Bengio**: $h_t = f(e(\text{hist}))$, $e(\text{hist}) = [e(y_{t-1}); e(y_{t-2}); \dots]$

RNN: $h_t = f(h_{t-1}, e(y_{t-1}))$ (implicit infinite context) **Vanilla**: $h_t = \sigma(W_1 h_{t-1} + W_2 e(y_{t-1}))$ **BPTT**: unroll through time, sum grads over timesteps

4. Word Embeddings

Encoding: **One-hot**: $v\in O(|V|)$, only word=1 **Bag-of-words**: pooled one-hot (sum/mean/max) **N-grams**: vectors huge—every combo needs slot **Pipeline**: Embedding → Pooling → NN → Softmax

Skip-gram: **Preprocess**: word-context pairs ($k\times C$ many), window k $p(c|w) = \frac{1}{Z(w)}\exp\left(e_{\text{wrd}(w)}\cdot e_{\text{ctx}(c)}\right)$, $O(2|V|k)$ params **Bilinear**: linear if all-but-one vars held constant **Similarity**: $\cos(u_i,u_j)$

5. CRF & POS Tagging

As Graph: Fully connected graph w/ POS-tag nodes per layer $\text{score}(\langle D,N,V,\dots\rangle, w) = \theta f(t,w)$ **score** (t,w)= unnormalized log-prob = $\sum_n$ trans+emit Problem: $O(|\mathcal{T}|^N)$ paths in normalizer

CRF Model: $p(t|w) = \frac{\exp(\text{score}(t,w))}{\sum_{t'\in\mathcal{T}^N}\exp(\text{score}(t',w))}$ **Decomposition**: $\text{score}(t,w) = \sum_{n=1}^N \text{score}(\langle t_{n-1},t_n\rangle, w, n)$ $p(t|w)\propto\prod_{n=1}^N \exp\{\text{score}(\langle t_{n-1},t_n\rangle,w)\}$

DP 推导: $O(|\mathcal{T}|^N)\to O(N|\mathcal{T}|^2)$: Goal: $Z = \sum_{t\in\mathcal{T}^N}\exp\text{score}(t,w)$ **Step1**: 可加分解 $\text{score} = \sum_n \text{score}_n$ **Step2**: $Z = \sum_t \exp\sum_n \text{score}_n = \sum_t\prod_n \exp\text{score}_n$ (exp) **Step3**: $= \sum_{t_1}\cdots\sum_{t_N}\prod_n \exp\text{score}_n$ (展开) **Step4**: $= \sum_{t_1}\exp\text{score}_1 \times \left(\sum_{t_2}\cdots\right)$ (distrib 把内层 sum 推进去) 若 **3-gram**: 依赖$t_{n-2}, t_{n-1}, t_n \to O(N|\mathcal{T}|^3)$

Forward Algorithm: $\alpha[0,t] = \exp(\text{score}(\text{BOS}\to t))$ (init w/ BOS trans) for $n=1,\dots,N-1$; for $t_n\in\mathcal{T}$: $\alpha[n,t_n] = \bigoplus_{t_{n-1}}\alpha[n-1,t_{n-1}]\otimes\exp(\text{score})$ return $\bigoplus_t \alpha[N-1,t]$ (sum last column!) 直觉: prefix 之和, 从 seq 开头走到当前状态的所有走法 score 总和

Backward Algorithm: $\forall t_N: \beta[N,t_N]\leftarrow 1$ for $n=N-1,\dots,0$; for $t_n\in\mathcal{T}$: $\beta[n,t_n]\leftarrow\bigoplus_{t_{n+1}}\exp(\text{score}_{n+1})\otimes\beta[n+1,t_{n+1}]$ return $\beta[0,\text{BOS}]$ (single value!) **Complexity**: $O(N|\mathcal{T}|^2)$

Fwd vs Bwd Asymmetry: **Init**: Bwd 直接1; Fwd 需 BOS 转移 **Term**: Bwd 返回$\beta[0,\text{BOS}]$单值; Fwd 需⊕整列 原因: BOS 显式存在, EOS 不显式处理

Viterbi Decoding: $\delta[n,t] = \max_{t_{n-1}}[\delta[n-1,t_{n-1}] + \text{score}(t_{n-1},t)]$ 每步枚举t和t_{n-1}→$|\mathcal{T}|^2$种 trans **Backtrack**: 存argmax指针 bp, 从$\arg\max_t\delta[N,t]$回溯

Common Semirings:

Name	$\mathbb{K}$	$\oplus$	$\otimes$	$\mathbf{0}$	$\mathbf{1}$	用途
Real	$\mathbb{R}_{\ge 0}$	$+$	$\times$	0	1	Z partition
Viterbi	$\mathbb{R}\cup\{-\infty\}$	max	$+$	$-\infty$	0	最优 path
Log	$\mathbb{R}\cup\{\pm\infty\}$	lse	$+$	$-\infty$	0	$\log Z$
Boolean	$\{0,1\}$	$\vee$	$\wedge$	0	1	可达性
Counting	$\mathbb{N}$	$+$	$\times$	0	1	路径数
Tropical	$\mathbb{R}\cup\{\infty\}$	min	$+$	∞	0	最短路

Semiring Definition: $\langle\mathbb{K},\oplus,\otimes,\mathbf{0},\mathbf{1}\rangle$ where:
1. $(\mathbb{K},\oplus,\mathbf{0})$: **comm monoid** (assoc+comm+identity)
2. $(\mathbb{K},\otimes,\mathbf{1})$: **monoid** (assoc+identity)
3. **Distrib**: $(x\oplus y)\otimes z = (x\otimes z)\oplus(y\otimes z)$
4. **Annihilator**: $\mathbf{0}\otimes x = x\otimes\mathbf{0} = \mathbf{0}$

Semiring 意义: ⊕: 分治 (split points 合并, OR/MAX/+) ⊗: 连接 (左右子树组合, AND/×/+) **0**: 吸收元, 消除invalid; **1**: 单位元, null 不破坏

Monoid 判定:
1. **Closure**: $a\otimes b\in\mathbb{K}$
2. **Assoc**: $(a\otimes b)\otimes c = a\otimes(b\otimes c)$
3. **Identity**: $\exists e: a\otimes e = e\otimes a = a$

Kleene Star: $a^* = \bigoplus_{n=0}^\infty a^{\otimes n} = \mathbf{1}\oplus a\otimes a^*$ Real 上 $|a|<1$: $a^* = \frac{1}{1-a}$ (geometric series) Tropical: $a^*=0$ if $a\ge 0$ (正环不帮助) 用于 globally normalized LM

6. CFG Parsing

Constituents: Multi-word units as single unit **Tests**: Pronoun substitution, Clefting, Answer ellipsis Ambiguity: PP attachment, modifier scope

CFG Definition: $G = \langle\mathcal{N},\mathcal{S},\Sigma,\mathcal{R}\rangle$ Non-terminals, start symbol, terminals, production rules **CNF**: $N_1\to N_2N_3$ or $N\to a$; $O(4^N)$ trees (Catalan)

Weighted CFG: **Global**: $p(t) = \frac{1}{Z}\prod_{r\in t}\exp(\text{score}(r))$ $Z = \sum_{t'\in\mathcal{T}}\prod_{r'}\exp(\text{score}(r'))$ (可能∞!) **Probabilistic**: local norm $\sum_k p(\alpha_k|N) = 1$

CKY Chart 索引: Position 在 words 之间: $0|w_1|1|w_2|2|\dots|N$ Chart$[i,k,X]$: span $[i,k)$覆盖 $w_i,\dots,w_{k-1}$ 长度: $k-i$; 对角线: $k-i=1$ (单词) **Fill order**: 按 span 长度递增($\ell=1,2,\dots,N$) 同一长度内任意顺序 (topo order 自由度) **Goal**: Chart$[0,N,S]$

CKY algo: $O(N^3|R|)$, needs CNF **Terminal**: $C[i,i+1,X] = \exp\text{score}(X\to w_i)$ for $X\to w_i\in\mathcal{R}$ **Binary**: for span$=2,\dots,N$; for $i=1,\dots,N-$span: $k\leftarrow i+$span; for $j=i+1,\dots,k-1$; for $X\to YZ\in\mathcal{R}$: $C[i,k,X]\oplus\exp\{\text{score}\}\otimes C[i,j,Y]\otimes C[j,k,Z]$

CKY Chart 3×3 Example: Sentence: $w_1w_2w_3$

	1	2	3
0	$C[0,1]$	$C[0,2]$	$C[0,3]$←**goal**
1		$C[1,2]$	$C[1,3]$
2			$C[2,3]$

Fill: diag first, then by span length

7. Dependency Parsing

Dependency Tree: Directed spanning tree, root degree 1 **Constraints**: Single head; Connected; Acyclic **Projective**: arcs 不交叉 (嵌套/并列) → CKY 可用 **Non-projective**: arcs 可交叉 → 必须用 CLE/MTT # spanning trees: $O((n-1)^{n-2})$

Edge-Factored Model: **Arc-factored**: 每条边独立打分, 树 score=边 score之和 优点: global优化分解为local边决策 局限: 无法捕捉 sibling/grandparent effects $\text{score}(t,w) = \sum_{(i\to j)\in t}\text{score}(i\to j,w) + \text{score}(r,w)$ $p(t|w) = \frac{1}{Z}\prod_{(i\to j)\in t}\exp(\text{score}(i,j,w))\exp(\text{score}(r,w))$

CLE 关键步骤: **Goal**: max spanning arborescence (directed MST)
1. For each node v, pick max incoming edge
2. If no cycle → done (it's a tree)
3. If cycle → **contract** cycle to supernode
4. **Reweight**: $\omega'(u\to v) = \omega(u\to v) - \omega_{\text{in-cycle}(v)}$
5. Recursively solve contracted graph
6. **Expand**: break cycle at min-loss edge

Complexity: $O(N^2)$ or $O(E+N\log N)$

Cayley Formula: 无向K_n: n^{n-2}棵 spanning trees 有向+固定 **root**: n^{n-2}棵 arborescences 有向+任意 **root**: $n\times n^{n-2} = n^{n-1}$棵

Graph Laplacian L: $L_{ij} = \begin{cases}\text{Degree}(i) & i=j \text{ (对角线)}\\ -1 & i\ne j\wedge i\sim j \text{ (有边)}\\ 0 & \text{otherwise}\end{cases}$ **trick**: 只看非对角−1判断边存在 **MTT**: #spanning trees = $\det(\hat L)$ (any minor)

Weighted Laplacian (MTT): $A_{ij} = \exp(\text{score}(i\to j))$, $\rho_j = \exp(\text{score}(j,w))$ $L_{ij} = \begin{cases}\rho_j & i=1 \text{ (root row)}\\ \sum_{k\ne j}A_{kj} & i=j \text{ (in-degree)}\\ -A_{ij} & \text{else}\end{cases}$ $Z = \det(L)$, 复杂度$O(n^3)$

Root Constraint: CLE base 允许多 root outgoing arcs **Naive**: 对每条 root arc 分别运行 CLE → $O(N\cdot\text{CLE})$ **Clever** (Gabow): swap score=next-best - current 删除 swap score 最小的多余 root edge

MTT vs CLE: [维度], [**MTT**], [**CLE**], [目标], [$Z = \sum_t\exp(\text{score})$], [$t^* = \arg\max$], [算法], [$\det(\hat L)$], [Greedy+Contract], [复杂度], [$O(N^3)$], [$O(N^2)$],

8. Semantic Parsing

Syntax vs Semantics: **Syntax**: structural org (parse tree) **Semantics**: underlying meaning **Logical form**: quantifiers, vars, boolean, predicates **Compositionality**: meaning of whole = fn of parts

Lambda Calculus: **Terms**: 变量x; 抽象$\lambda x.M$; 应用(MN) **β-reduction**: $(\lambda x.M)N \xrightarrow[\beta]{} M[x:=N]$ **α-conversion**: 重命名 bound 变量避免 capture **β-infinity**: $F = \lambda x((xx)x)$, $FF = \dots$不终止

β-reduction 步骤:
1. 找到$(\lambda x.M)N$形式的 redex
2. 在M中找所有被该λx绑定的x
3. 将这些x替换为N

注意: 可能需先α-convert 避免变量捕获!

Free vs Bound Variables: $\text{FV}(x) = \{x\}$; $\text{FV}(\lambda x.M) = \text{FV}(M) - \{x\}$ $\text{FV}(MN) = \text{FV}(M)\cup\text{FV}(N)$ **Bound**: 在某λ的 scope 内 **Free**: 不在任何 abstraction 的 scope 内

Combinatory Logic: $\boldsymbol{I}x = x$; $\boldsymbol{K}xy = x$; $\boldsymbol{S}xyz = xz(yz)$ $\boldsymbol{B}xyz = x(yz)$ (comp); $\boldsymbol{C}xyz = xzy$ (flip)

$\boldsymbol{T}xy = yx$ (type-raising) $\boldsymbol{I} = \boldsymbol{SKK}$ (S,K 构成 complete basis)

CCG Rules: **Application**: $X/Y\ Y \Rightarrow X$ (>前向); $Y\ X\setminus Y \Rightarrow X$ (<后向) **Composition**: $X/Y\ Y/Z \Rightarrow X/Z$ ($\boldsymbol{B}_>$) **Type-raising**: $X \Rightarrow T/(T\setminus X)$ ($\boldsymbol{T}_>$) rules 是 universal, language-specific 全在 lexicon

CCG Category 直觉: $S\setminus$NP: 左边要 NP → 产出 S (intransitive) $(S\setminus\text{NP})/\text{NP}$: 右边要 NP → $S\setminus$NP (transitive) **Slash** 方向: / 向右找 arg; \ 向左找 arg

Derivation with Semantics: Lexicon:
- Mary : NP : Mary
- likes : $(S\setminus\text{NP})/$ NP : $\lambda y.\lambda x.$ Likes(x,y)
- John : NP : John

Parse "Mary likes John":
```
Mary         likes                      John
NP:Mary (S\NP)/NP:λy.λx.Likes(x,y)  NP:John
                  ---------------------------------- >
                          S\NP:λx.Likes(x,John)
--------------------------------------------------- <
                   S:Likes(Mary,John)
```

LIG 构造策略: 问题: CFG 无法"计数" ($a^nb^nc^n$ 中n相等) **LIG**: 用 stack 记录计数信息 策略 **1**: 两端向中间—先生成首尾, 再生成中间 策略 **2**: 左向右—前半部分 push, 后半部分 pop **Example** $a^nb^nc^nd^n$: $S[\sigma] \to aS[f\sigma]d$; $S[\sigma] \to T[\sigma]\ T[f\sigma] \to bT[\sigma]c$; $T[] \to \varepsilon$

FOL Translation: ∀配⇒:全称限定条件; ∃配∧: 存在某具体对象; 否则∃配⇒:往往荒谬; ∨配∧:要求满足多个条件.

9. WFST & Lehmann

Transducer Def: $T = \langle Q,\Sigma,\Omega,\lambda,\rho,\delta\rangle$ Q: states; Σ: input; Ω: output $\lambda: Q\to\mathbb{R}$: initial; $\rho: Q\to\mathbb{R}$: final $\delta: Q\times(\Sigma\cup\varepsilon)\times(\Omega\cup\varepsilon)\times Q\to\mathbb{R}$ **ε-transition**: no input/output consumed

FSA vs FST: **WFSA** (单带): read only, score$(\pi) = \sum_n$ score(τ_n) **WFST** (双带): read input + write output **Unambiguous**: $|\Pi(x,y)|\le 1$ **Ambiguous**: $|\Pi(x,y)|>1 \to$ need semiring

Path Score: score$(\pi) = \lambda(q_{\text{start}}) + \sum_{n=1}^{|\pi|}$ score$(\tau_n) + \rho(q_{\text{end}})$ $p(y|x) = \frac{1}{Z}\sum_{\pi\in\Pi(x,y)}\exp(\text{score}(\pi))$ $Z = \sum_{y'\in\Omega^*}\sum_{\pi'}\exp(\text{score}(\pi'))$ (infinite!)

Matrix Mult View: $C = A\otimes B$: $C_{ij} = \bigoplus_k(A_{ik}\otimes B_{kj})$ **Tropical**: $C_{ij} = \min_k(A_{ik}+B_{kj})$ **Inside**: $C_{ij} = \sum_k(A_{ik}\times B_{kj})$ Naive W^N: $O(N^4)$ → Lehmann fixes to $O(N^3)$

Lehmann 递推直觉: $\boldsymbol{R}_{ik}^{(j)}$: 从 q_i 到 q_k, 仅经过 $\{q_1,...,q_j\}$的 paths 总权分解:

$$\boldsymbol{R}_{ik}^{(j)} = \boldsymbol{R}_{ik}^{(j-1)} \oplus \left(\boldsymbol{R}_{ij}^{(j-1)} \otimes \left(\boldsymbol{R}_{jj}^{(j-1)}\right)^* \otimes \boldsymbol{R}_{jk}^{(j-1)}\right)$$

不经q_j + (到q_j + 在q_j循环任意次 + 离开q_j)
$Z = \bigoplus_{i,k\in Q}\lambda(q_i)\otimes\boldsymbol{R}_{ik}\otimes\rho(q_k)$
λ: initial weights ρ: final weights $\boldsymbol{R}_{ik}$: Lehmann 算出的 all-paths 权重

Floyd-Warshall: **Key**: allow 中间 node k incrementally
$\text{dist}_k[i][j] = \min(\text{dist}_{k-1}[i][j], \text{dist}_{k-1}[i][k] + \text{dist}_{k-1}[k][j])$
Runtime: $O(N^3)$ **FW** 是 Lehmann 在 Tropical 的特例 ($a^*=0$ 循环不帮助)

Lehmann algo: **Generalized FW** for any closed semiring:

$$W_{ij}^{(k)} = W_{ij}^{(k-1)} \oplus W_{ik}^{(k-1)} \otimes \left(W_{kk}^{(k-1)}\right)^* \otimes W_{kj}^{(k-1)}$$

定义: $\boldsymbol{R}_{ik}^{(j)}$=从$q_i$到$q_k$, 仅经过$\{q_1,...,q_j\}$的 paths 的 semiring-sum 直觉: 经过$\{1,...,j\}$的 paths = 不经j ⊕ 经j后者分解: $i\to j$(不经j) + j上 cycles + $j\to k$(不经j) Runtime: $O(|Q|^3)$

Pathsum & Z: $Z(\mathcal{T}) = \bigoplus_{i,k\in Q}\lambda(q_i)\otimes\boldsymbol{R}_{ik}\otimes\rho(q_k)$
$Z = \alpha^\top\left(\bigoplus_{\omega\in\Sigma^*}W^{(\omega)}\right)^*\beta$ **Why Lehmann?** Direct sum over infinite paths impossible

Composition: $\mathcal{T}(x,y) = \bigoplus_{z\in\Omega^*}\mathcal{T}_1(x,z)\otimes\mathcal{T}_2(z,y)$
Transliteration: 3 transducers cascade $\mathcal{T}_x\circ\mathcal{T}_\theta\circ\mathcal{T}_y$

Acyclic WFSA Backward: 前提: DAG 可做 topological sort
1. 按 reverse topo order 遍历 nodes $q_M,...,q_1$
2. $\beta[q_m] \leftarrow \rho(q_m)\oplus\bigoplus_{(q_m,a,w,q')\in\delta} w\otimes\beta[q']$
3. return $\bigoplus_{q\in I}\lambda(q)\otimes\beta[q]$

Complexity: $O(|Q|+|\delta|)$ (linear!)

10. Transformers & MT

Seq2Seq: z = encoder(x), $y|x\sim$ decoder(z) $p(y|x) = \prod_{t=1}^T p(y_t|x,y_1,...,y_{t-1})$ **Information Bottleneck**: z fixed-length → Attention 解决

Attention: $\alpha^TV = \sum_i\alpha_iv_i^T$ (soft retrieval) α_i = sftm(score(q,k_i)) $K=V=H^{(e)}$, $q_t = h_t^{(d)}$, $c = \alpha^TV$

Self-Attention: $\boldsymbol{Q} = \boldsymbol{X}\boldsymbol{W}_Q$, $\boldsymbol{K} = \boldsymbol{X}\boldsymbol{W}_K$, $\boldsymbol{V} = \boldsymbol{X}\boldsymbol{W}_V$ SelfAtt = sftm$\left(\frac{\boldsymbol{QK}^\top}{\sqrt{d_k}}\right)\boldsymbol{V}$ $\sqrt{\boldsymbol{d_k}}$: 防止点积过大导致 softmax 饱和; σ^2. **Complexity**: $O(nd^2+dn^2)$ Permutation Equivariance: 若 f 是 permutation equivariant, 则对任意 permutation π, $f(\pi(X)) = \pi(f(X))$, 若 $\boldsymbol{Q}$ fixed (如常数矩阵), 则 attention permutation invariant.即 打乱输入顺序, 输出以相同方式打乱. 设 $\boldsymbol{P}$ 是 permutation matrix, 则: Attn$(\boldsymbol{PX})$ = sftm$\left(\frac{1}{\sqrt d}(\boldsymbol{PXW}_Q)(\boldsymbol{PXW}_K)^\top\right)(\boldsymbol{PXW}_V)$ = sftm$\left(\boldsymbol{PQK}^\top\frac{\boldsymbol{P}^\top}{\sqrt d}\right)\boldsymbol{PV} = \boldsymbol{P}$ sftm$\left(\boldsymbol{Q}\frac{\boldsymbol{K}^\top}{\sqrt d}\right)\boldsymbol{V} = \boldsymbol{P}$ Attn$(\boldsymbol{X})$

Positional Encoding: $\boldsymbol{P}_{p,2i} = \sin(p/10000^{2i/d})$ $\boldsymbol{P}_{p,2i+1} = \cos(p/10000^{2i/d})$ motiv: Transformer 无 recurrence, 无法区分位置

Encoder-Decoder 架构: **Encoder**: +$\boldsymbol{P}$ → MHSA → + → LN → MLP → + → LN **Decoder**: +masked self-attn + cross-attn **Masked**: 只 attend 到左边 positions (causal) **Cross-attn**: Q来自 decoder, K,V来自 encoder **Residual**: x + Layer(x) 缓解 vanishing gradient

Decoding Strategies: $y^* = \arg\max_{y\in\mathcal{Y}}$ score(x,y) W/o assumptions: $O(|\Sigma|^{n_{\max}})$ paths **Greedy**: 每步 arg max (次优, 快) **Beam**: 保持k-best candidates **Nucleus/Top-p**: 从累积 prob≥ p的 tokens 中 sample **Temperature**: $T<1$ sharper; $T>1$ uniform **Eval**: BLEU (n-gram overlap), METEOR

MT Pipeline:
1. **Tokenize**: subword (BPE/WordPiece)
2. **Embed**: token→vector + positional
3. **Encode**: Transformer encoder
4. **Decode**: autoregressive, $p(y_n|y_{<n},\boldsymbol{z})$
5. **Search**: beam/nucleus sampling

Train: MLE, $-\sum\log p(y_n|y_{<n},\boldsymbol{x})$

11. Modeling Choices

Prob vs Non-Prob: **Prob**: leverage prob theory, needs assumptions CRF, RNN, N-gram models **Non-Prob**: interpretable, uncertainty unclear Perceptron, SVM, CFG rules

Disc vs Generative: **Discriminative**: model boundary $p(y|x)$ **Generative**: model own dist $p(x,y)$

Local vs Global Norm: **Local**: efficient train, biased predictions **Global**: needs Z, unbiased Independence assumptions control complexity

Regularization: **LogLoss**: $\ell(y,y') = \log\left(1+e^{-y\cdot y'}\right)$ **Exp-Loss**: $\ell(y,y') = e^{-y\cdot y'}$ **L1/L2**: weight penalties (Laplace/Gaussian prior)

Evaluation Metrics: **Prec**: $P_{\text{true}}/P_{\text{all}}$; **Recall**: $P_{\text{true}}/(P_{\text{true}}+N_{\text{false}})$ **Acc**: $(P_{\text{true}}+N_{\text{true}})/N$ **F-score**: $((1+\beta^2)(\text{prec}\cdot\text{recall}))/(\beta^2\text{prec}+\text{recall})$

Statistical Tests: $p = 2\min(P(T\ge t|H_0), P(T\le t|H_0))$; Rej if $p<\alpha$ **Power**: $P(\text{reject } H_0|H_1)$ **Multiple tests**: $P(|\text{FalseRej}|>0) = 1-(1-\alpha)^K$ **Bonferroni**: $\alpha^* = \alpha/K$ **McNemar**: $\chi^2 = \frac{(b-c)^2}{b+c}\sim\chi_1^2$

12. Bias & Fairness &Eval

Bias Sources: **Labeling**: reproduce annotator bias **Sample selection**: training fits certain profile **Task definition**: excludes certain groups **Imbalanced test**: loss ignores minorities

BLEU Score: BLEU = BP × exp$\left(\sum_{n=1}^N w_n\log p_n\right)$ p_n: n-gram precision (clipped count) BP = $\begin{cases}1 & c>r\\ e^{1-r/c} & \text{otherwise}\end{cases}$ c=候选长度, r=参考长度, $w_n = 1/N$ **Clipped**: count$_{\text{clip}}$ = min$\left(\text{count}_{\text{pred}}, \max_{\text{ref}}\text{count}_{\text{ref}}\right)$ 防止重复词刷分

Model Taxonomy: **Probabilistic**: 建模 $p(Y|X)$或 $p(X,Y)$
- **Discriminative**: 直接$p(Y|X)$ (LogReg, CRF)
- **Generative**: joint $p(X,Y) = p(Y)p(X|Y)$ (N-gram, HMM)

Non-Prob: Learned (SVM, MLP) / Handcrafted (CFG)

Confusion Matrix Metrics:

	Pred +	Pred −
Actual +	TP	FN
Actual −	FP	TN

Prec = TP/(TP + FP); Recall = TP/(TP + FN) $\text{F}_1 = 2\cdot(\text{Prec}\cdot\text{Recall})/(\text{Prec}+\text{Recall})$ 为何不用 **Acc?** Class imbalance; 不同错误代价不同

K-Fold CV: 数据分K份, 每次取第k份为 test, 其余 train **Test set size**: N/K **Train set size**: $N\times(K-1)/K$ **Total models**: K **Nested CV**: Inner loop 调参, Outer loop 评估

McNemar's Test: 比较两个 classifiers 在同一数据集上表现

	B Correct	B Wrong
A Correct	n_{00}	n_{01}
A Wrong	n_{10}	n_{11}

$\chi^2 = \frac{(|n_{01}-n_{10}|-1)^2}{n_{01}+n_{10}}$ 只关注 disagreement cells n_{01},n_{10} 要求$n_{01}+n_{10}\ge 25$

Permutation Test:
1. 原始数据训练, 记录 performance P_0
2. Repeat $B\ge 1000$次: permute labels, 重训, 记录P_b
3. p-value ≈ fraction of $P_b\ge P_0$

tip: 若 labels 有信息, 原始模型应显著优于 permuted

补充

Edit Distance FSA: 状态: (i,e) 位置×编辑次数, 共$O(dN)$个 转移: 匹配$s_i\to(i+1,e)$; 插 $\Sigma\to(i,e+1)$; 删 $\varepsilon\to(i+1,e+1)$; 替$\Sigma\setminus s_i\to(i+1,e+1)$ 终态: $i=N$所有状态 口诀: 插读不动, 删 ε 跳, 替错跳

Semiring 速判: 先验$\mathbf{0}\oplus a = a$ min单位元=+∞ (非0/−∞) max单位元=−∞ **Kleene**: Real $a^* = \frac{1}{1-a}$; Bool $a^*=1$

BPTT: $\frac{\partial h_t}{\partial h_k} = \prod_i \text{diag}(\sigma')R$ $R^n = QD^nQ^{-1}$ $|\lambda|<1$→消失; $|\lambda|>1$→爆炸

FOL: ∀配⇒; ∃配∧ "所有 X 都 Y": $\forall x.X(x)\Rightarrow Y(x)$ "有些 X 是 Y": $\exists x.X(x)\wedge Y(x)$

β-reduce: $(\lambda x.M)N\to M[x:=N]$ $(\lambda x.(xx))(\lambda z.x) = (\lambda z.x)(\lambda z.x) = x$

Fwd vs Bwd 不对称性:

项目	Forward	Backward
初始化	$\alpha[0,t] = \exp(\text{score}(\text{BOS}\to t))$	$\beta[N,t] = \mathbf{1}$ 全 1
递推	$\alpha[n,t] = \bigoplus_{t'}\alpha[n-1,t']\otimes\exp$	$\beta[n,t] = \bigoplus_{t'}\exp\otimes\beta[n+1,t']$
终止	$\bigoplus_t\alpha[N,t]$ 需 sum	$\beta[0,\text{BOS}]$ 单值

原因: BOS 显式存在, EOS 隐式处理使用场景 **Forward**: 单独计算Z (partition function) **Backward**: 单独计算 suffix 概率 两者结合: 计算 marginals $p(t_n=t|\boldsymbol{w})$ $p(t_n=t|\boldsymbol{w}) = \frac{\alpha[n,t]\times\beta[n,t]}{Z}$

Quick Ref

Chain: $\frac{d}{dx}[f(g(x))] = f'(g)g'(x)$; Bauer: sum over all paths **Softmax**: $\exp(h_y)/\sum\exp(h_{y'})$; $T\to 0$ =argmax **Log-Linear**: $p(y|x) = \exp(\theta\cdot f)/Z$; MLE matches expected features **DP**: distrib 把$O(|T|^N)\to O(N|T|^2)$; 3-gram 则$O(N|T|^3)$ **Fwd/Bwd**: Fwd init BOS+sum last col; Bwd init **1**+single value **Viterbi**: max instead of sum + backpointer **CKY**: $O(N^3|R|)$; CNF; diag first, span 递增 **MTT**: $Z = \det(L)$ in $O(n^3)$; **CLE**: greedy+contract $O(n^2)$ **Lehmann**: $R^{(j)} = R^{(j-1)}\oplus R\otimes R^*\otimes R$; $O(|Q|^3)$ **Kleene**: Inside $\frac{1}{1-a}$; Tropical 0 if $a\ge 0$ **CCG**: $X/Y\ Y\Rightarrow X$ (>); $Y\ X\setminus Y\Rightarrow X$ (<) **β-reduce**: $(\lambda x.M)N\to M[x:=N]$; 先$\alpha$-convert 避免捕获 **Self-Attn**: sftm$\left(QK^T/\sqrt d\right)V$; $O(nd^2+dn^2)$ **Cayley**: 固定 root n^{n-2}; 任意 root n^{n-1}