LANGUAGE MODELING

Alphabet Σ finite, non-empty set of symbols Vocab words / tokens $\varepsilon \notin \Sigma$
String over Σ is a finite Sequence of alphabet
Kleene Closure $a^* = \bigoplus_{n=0}^{\infty} a^{\otimes n} = a^{\otimes 0} \oplus a^{\otimes 1} \oplus \bigoplus_{n=2}^{\infty} a^{\otimes n} = 1 \oplus a \otimes \bigoplus_{n=0}^{\infty} a^{\otimes n}$
Language Model Distribution over Σ^* Global Σ^* is infinite Sum over $y \in \Sigma^*$
Locally Normalized (Autoregressive) Collection of conditionals $p(y \mid \mathbf{y})$
String Probability $p(\mathbf{y}) = p(y_1 \mid \text{BOS}) \, p(y_2 \mid \text{BOS}\, y_1) \cdots p(y_N \mid \mathbf{y}_{<N}) \, p(\text{EOS} \mid \mathbf{y})$
One-hot Encoding Vocab V, $e(w) \in \mathbb{R}^{|V|}$ BagOfWords Sum up one-hot embeddings
n-grams Encode n-grams, $|V| \to |V|^n$ PreProcessing Tokenization / Lemmatization / Punct Removal
Skip Gram Sentencize and Tokenize Corpus, then Build Positive Samples with given Context window k. Results $N = O(k \cdot C)$, $e_{wrd} \in \mathbb{R}^d$ $e_{ctx}(c) \in \mathbb{R}^d$ log-bilinear model
$p(c \mid w) = \frac{1}{Z(w)} \exp(e_{wrd}(w) \cdot e_{ctx}(c))$ Train objective: Maximize NLLK
n-gram Assumption $p(y_t \mid \mathbf{y}_{<t}) = p(y_t \mid y_{t-n+1} \cdots y_{t-1})$ Context depends on prev n-1
↳ One conditional probability for each possible context $\sum_{j=0}^{n-1} |\Sigma|^j$
↳ $|\Sigma| - 1$ free parameters for each conditional probability distribution
Bengio's Model Estimate $p(y \mid \mathbf{y})$ more efficiently. Share parameters between contexts $y_{t-n+1} \cdots y_{t-1}$, Instead of treating words as discrete symbols, treat as embeddings

$$p(y_t \mid \mathbf{y}_{<t}) = \frac{\exp(v(y_t) \cdot h_t)}{\sum_{y' \in \Sigma} \exp(v(y') \cdot h_t)}$$

Word Embeddings Context "summary"
Similarity: Dot Prod between $v(y) \in \mathbb{R}^d$ $h_t \in \mathbb{R}^d$
Advantage: Naive Ngram No long-term dependencies Cons: Slow (Softmax Z $\forall$ conditionals)
Solve Curse of Dimensionality 1. Word Embeddings 2. MLP to Combine Embeddings 3. Softmax
RNN $h_t = \sigma(U h_{t-1} + V u(y_{t-1}) + b_h)$ $p(y_t \mid \mathbf{y}_{<t}) = \exp(v(y_t) \cdot \overline{h_t}) / \sum_{y' \in \Sigma} \exp(v(y') \cdot h_t)$

CONDITIONAL RANDOM FIELDS

$$p(\mathbf{t} \mid w) = \frac{\exp\{\sum_{n=1}^{N} \text{score}(\langle t_{n-1}, t_n \rangle, w)\}}{\sum_{t \in \mathcal{T}^N} \exp\{\sum_{n=1}^{N} \text{score}(\langle t_{n-1}, t_n \rangle, w)\}}$$

Naive Z: $O(|\mathcal{T}|^N)$
With Distributivity: Linear # of terms

$$Z = \sum_{t_0 \in \mathcal{T}} \left(\exp\{\text{score}(\langle t_0, t_1 \rangle, w)\} \times \cdots \times \sum_{t_{N-1} \in \mathcal{T}} \left(\exp\{\text{score}(\langle t_{N-2}, t_{N-1} \rangle, w)\} \times \sum_{t_N \in \mathcal{T}} \exp\{\text{score}(\langle t_{N-1}, t_N \rangle, w)\} \cdots \right) \right)$$

Viterbi Algorithm Choose Semiring $R = \langle \mathbb{R}^+, +, \times, 0, 1 \rangle$ and compute
$\beta(\vec{w}, t_N) = 1$ Works because of the Distributive Property $O(T^2 N)$
for $n = N-1, \ldots, 0$: Correctness: Just Prove Semiring Axioms

$$\beta(w, t_n) = \bigoplus_{t_{n+1} \in \mathcal{T}} \exp\{\text{score}\langle t_n, t_{n+1} \rangle, w\} \otimes \beta(w, t_{n+1})$$

$0 \oplus a =$
Semirings $R = \langle A, \oplus, \otimes, 0, 1 \rangle$ Associativity $(a \oplus b) \oplus c = a \oplus (b \oplus c)$ Identity $a \oplus 0$
Commutativity $a \oplus b = b \oplus a$. $\oplus$ is a Commutative Monoid $\otimes$ is a Monoid
Distributivity $(a \oplus b) \otimes c = a \otimes c + b \otimes c$ and $a \otimes (b \oplus c) = a \otimes b \oplus a \otimes c$ Axiom 4 $0 \otimes a = 0$

Dijkstras $O(|w||T|^2 \log|w||T|)$ Emission $t \to w$ Transition $t \mapsto t$
Semirings BOOLEAN $\langle \{0,1\}, \vee, \wedge, 0, 1 \rangle$ logical deduction, recognition
Viterbi $\langle [0,1], \max, \times, 0, 1 \rangle$ prob. of best derivation Shortest
Inside $\langle \mathbb{R}^+ \cup \{+\infty\}, +, \times, 0, 1 \rangle$ prob. of a string Real $\langle \mathbb{R} \cup \{+\infty\}, +, \times, 0, 1 \rangle$ distance
TROPICAL $\langle \mathbb{R}^+ \cup \{+\infty\}, \min, +, +\infty, 0 \rangle$ Shortest Distance with Non-Negative Weights
Counting $\langle \mathbb{N}, +, \times, 0, 1 \rangle$ Number of paths

WEIGHTED FINITE STATE AUTOMATA / TRANSDUCERS (WFSA, WFST)

WFST $\mathcal{T}$ over $\mathcal{W} = (\mathbb{K}, \oplus, \otimes, \bar{0}, \bar{1})$ is an 8-tuple $(\Sigma, \Delta, Q, I, F, \delta, \lambda, \rho)$ Where
Σ is a finite input alphabet Δ is a finite output alphabet Q is a finite set of States
$I \subseteq Q$ is a set of Initial states $F \subseteq Q$ is a set of Final States $\delta \subseteq Q \times (\Sigma \cup \{\varepsilon\}) \times (\Delta \cup \{\varepsilon\}) \times \mathbb{K} \times Q$ is a finite Multi-set of transitions $\lambda : Q \to \mathbb{K}$ an init weighting function over the set of states Q $\rho : Q \to \mathbb{K}$ a final weighting function over set of states Q
$I = \{q \in Q \mid \lambda(q) \neq \bar{0}\}$ and $F = \{q \in Q \mid \rho(q) \neq \bar{0}\}$
Lehmann's Algorithm Solves all pairs shortest paths given no negative cycles $O(|V|^3)$
$W \leftarrow |Q| \times |Q|$ matrix over a closed semiring
$R^{(0)} \leftarrow W$
for $j \leftarrow 1$ up to $|Q|$:
  for $i \leftarrow 1$ up to $|Q|$:
    for $k \leftarrow 1$ up to $|Q|$:
      $R_{ik}^{(j)} \leftarrow R_{ik}^{(j-1)} \oplus (R_{jj}^{(j-1)})^* \otimes R_{jk}^{(j-1)}$
return $I \oplus R^{(|Q|)}$

$$(I+M)^k = \bigoplus_{n=0}^{k} \binom{k}{n} I^{k-n} \otimes M^n = \bigoplus_{n=0}^{k} \binom{k}{n} M^n$$

$$\|A\|_2 = \sup_{x \neq 0} \|Ax\|_2 / \|x\|_2 = \sigma_1 = \sigma_{\max}(A)$$

SVD $A = U \Sigma V^T$ Converges if $\sigma_{\max}(A) < 1$
$\to A^*$ Upperbound on Truncation Error

$$\|A^* - \sum_{n=0}^{k} A^n\|_2 \leq \|\sum_{n=k+1}^{\infty} A^n\|_2 \leq \sum_{n=k+1}^{\infty} \sigma_{\max}(A)^n = \frac{\sigma_{\max}(A)^{k+1}}{1 - \sigma_{\max}(A)}$$

Composition Let $T_1 : U \to V$ and $T_2 : V \to W$, $T_1 \circ T_2 : U \to W$

$$T(x, y) = \bigoplus_{z \in \Delta^*} T_1(x, z) \otimes T_2(z, y)$$

Remember to Relabel ε transitions

CONSTITUENCY PARSING

CFG $G = \langle N, S, \Sigma, R \rangle$ N {nonterminal Symbols} $\{N_1, N_2, N_3, N_4, \ldots\}$ S Start symbol
Σ Alphabet of terminal Symbols a_1 R {production Rules} $N \to \alpha$ $N \in \mathcal{N}$ $\alpha \in (\mathcal{N} \cup \Sigma)^*$
Chomsky Normal Form Grammar in CNF is all productions form of $N_1 \to N_2 N_3$ $N \to a$
Locally Normalized Distribution over each production is a valid probability distribution
Weighted CFG log-linear models over trees in grammar, structured Softmax
$p(t) = \frac{1}{Z} \prod_{r \in t} \exp\{\text{score}(r)\}$ $Z = \sum_{t' \in \mathcal{T}} \prod_{r \in t'} \exp\{\text{score}(r)\}$ $\mathcal{T}$ is infinite!
Divergence $S \to S(1 \cdot 0)$ $S \to a(1 \cdot 0)$, infinite sum because each summand is 1
CKY (max, x) Semiring → Score of one-best Parse, (Bool) → Checks if string in Grammar
Compute $Z(s)$ in $O(N^3(|R|)$ → $\nabla \log Z(s)$ in $O(N^3(|R|)$ Backprop!

DEPENDENCY PARSING

Projective No overlapping Arcs **Non-projective** Overlapping Arcs

Probability Distr Non-Projective Z Naively $O(N^N)$ Spanning Trees N^{N-2} + root Constraint $(N-1)^{N-2}$

Edge-factored Assumption $p(t|w) = \frac{1}{Z}\prod_{(i\to j)\in t}\exp\{score(i,j,w)\}\exp\{score(r,w)\}$

$Z = \sum_{t'\in T(w)}\prod_{(i\to j)\in w}\exp\{score(i,j,w)\}\exp\{score(r,w)\}$

MTT For an undirected unweighted graph G with N vertices, let L be the graph Laplacian $L_{ij} = \begin{cases} -A_{ij} & i\neq j \\ \rho_j + \sum_{k\neq i} A_{ik} & i=j \end{cases}$ Tuttle Generalized to Directed Case. Find Z in $O(N^3)$ $Z = |L|$ ρ_j: Root score

CLE ρ pseudoroot → Incorporate root scores. Need node that can reach all other nodes in the graph. Greedy Select best incoming edge to each node except ρ, if cycle Contract and Repeat. Otherwise, Constrain Root to only 1 edge by Comparing cost.

$N_{DT}(G') = NN_T(G)$, at root $1 = NN^{-2}$ $(\hat{L}_i - \lambda I)v^{(p,q)} = 0$ $\sum_i \xi_i u_i \otimes v_i \geq 0$

$x^T A x \geq 0$, square ⇒ PSD Contract: Update each enter edge to Cycle by adding the weights including enter edge but not including edge we break. $O(EN) = O(N^3)$

Tarjan → $O(E + N\log N)$

SEQ2SEQ MODELS

y at Inference, generate y_1 according to $p(\cdot|x)$, y_2 according to $p(\cdot|x, y_1)$

Attention weights $\alpha_i = softmax(score(q,k))_i$ Context $c = \sum_i \alpha_i v_i = \alpha V$

$k_i = v_i = h_i^{(e)}$ Vector representation [hidden state] produced by encoder at position i $q_t = h_t^{(d)}$ hidden state produced by decoder position t

$K = V = H^{(e)}$ Vertically stacked encoder vector representations

$\alpha = softmax(score(q,K))$ $MHSA(Z) = \sum_{h\in[N_h]} softmax(A^{(h)})ZW_v^{(h)}W_{out} + b_{out}$

Instead of encoding all input into a single context vector, pay variable attention depending on output generation step. MHSA: learn multiple sets of attention weights for same input then concat.

Scaled: $score(q_t, h_t) = softmax((q_t^T h_t)/\sqrt{d_k})V$ **PE** Encode order of words

RESIDUAL CONNECTIONS Mitigate Vanishing Gradients.

LayerNorm Helps with internal covariate shift, normalize individual Layer inputs INPUT → PE → N X [MHSA, LayerNorm, FF, LayerNorm]

TRANSFORMER OUTPUT → PE → N X [MHSA, LN, MHSA, LN, FF, LN] - Linear - Softm

DECODING $y^* = \arg\max score(x,y)$. Beam Search, Sampling where words are sampled iteratively from $p(y_t|x, y_{<t})$ Nucleus Sampling, only top words p%

SEMANTIC PARSING

Compositionality Meaning of Complex expression is a function of the meanings of that expressions constituent parts. → Build Models over Components

$\lambda x.f(x)$ Function that takes in x as input and F(x) as output.

α-Conversion $\lambda x.(xx)y \to \lambda t.(tt)y$ Rename var in abstraction with all bound vars

β-Reduction Apply lambda term to another $(\lambda x.\lambda y.(x((\lambda x.x\ x)\ y))\ z)$

$\to \lambda y.z((\lambda x.x\ z)y))$

AXES OF MODELING

Implications of Structural Constraints Determine complexity of the model.

A model with too many constraints (too simple) might underfit the data.

Too few constraints (too complex) → overfit. Domain specific structures, sequential data.

Cost/Benifits of Independence Assumptions simplifies computation and model understanding. Benifical with high dimensional data. Loss of Accuracy.

Regularization Increases models generization performance.

Considerations Data Balance, Robustness to Noise, Interpretability, Transferability

Inflated Confidence Overfitting to training data, Data Leakage, Improper Cross Val, Ignoring Model assumptions, Selective Reporting of Results.

OTHER IMPORTANT STUFF

Log-Linear Models Very general family of probability distributions that can be defined as $p(y|x,\theta) = \frac{1}{Z(\theta)}\exp(\theta\cdot f(x,y))$ If we take log → Linear function

Can be thought as dot-product followed by the Softmax

Softmax T: Non-negative temperature. $T\to\infty$ uniform (Maximum Entropy)

$T\to 0$ (Annealing) all mass placed on maximum (Minimum Entropy)

Exponential Family $p(x|\theta) = \frac{1}{Z(\theta)}h(x)\exp(\theta,\phi(x))$ generalizes Softmax

FF Neural Network Connection between nodes dont form Cycles.

$\nabla L(\theta) = -\sum_{n=1}^{N}\log p(y_n|x_n,\theta) = \underbrace{-\sum_{n=1}^{N} f(x_n,y)}_{\text{observed features counts}} - \underbrace{\sum_{n=1}^{N}\sum_{y\in Y} p(y'|x_n,\theta)\cdot f(x_n,y')}_{\text{= Expected Feature Counts}}$

Expectation Matching $\nabla L(\theta) = 0$

$p(w_{:k}|X) = \sum_{Y,Z\in N} p(X\to YZ)\left[\sum_{j=i}^{k-1} P_{inside}(w_{i:j}|Y)P_{PRE}(w_{j+1:k}|Z) + P_{PRE}(w_{i:k}|Y)\right]$

$M^* = I + M + M^2 + \ldots = \sum_{n=0}^{\infty} M^n = I + M\sum_{n=0}^{\infty} M^n = I + MM^* = (I-M)^{-1}$

Matrix Inversion $O(|N|^3) \Rightarrow P_{LC}(Y,Z|X) \in O(|N|^4)$ $P_{inside}(w_{i:j}|Y)$ by CKY $\in O(N^3|R_1|^3) = O(N^3|N|^3)$ $P_{pre}(w_k|X) \in O(|N|^2 N)$ $P_{PRE}(w) \in O(N^3|N|^3 + |N|^4)$

All prefix probability → N times $\in O(N^4|N|^3 + N|N|^4)$