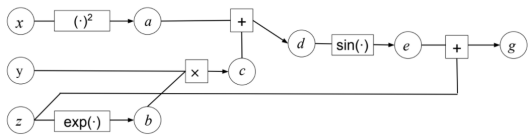
1 Backpropagation

Goal

Compute function gradient in the complexity of computing the function value.

Computational Graph with Hyperedges

$$f(x,y,z) = z + \sin(x^2 + y \times \exp(z))$$



2 Log-linear Models

Definition

$$p(y \mid x, \theta) = \frac{\exp(\theta \cdot f(x,y))}{\sum_{y' \in \mathcal{Y}} \exp(\theta \cdot f(x,y'))}$$

The gradient of MLE loss $L(\theta) = \sum_n \log p(y_n \mid x_n, \theta)$ is $\frac{\partial L(\theta)}{\partial \theta_k} = \sum_n f_k(x_n, y_n) - \sum_n \sum_{y'} p(y' \mid x_n, \theta) f_k(x_n, y')$, which is the observed "feature count" minus the expected "feature count".

Exponential Family

$$p(x \mid \theta) = \frac{h(x)}{Z(\theta)} \exp(\theta \cdot \phi(x))$$

It maximizes entropy under the constraint of a fixed expectation.

3 Sentiment Analysis with MLP

Goal

Predict whether a sentence is positive or negative.

Skip-gram Model (word2vec)

1. Preprocess: Given window size k and corpus size C, find all pairs with a focus word w and a context word c which locates in the window centered around the focus word. The complexity is $O(k \cdot C)$.
2. The model: $p(c \mid w) = \frac{\exp(e_{\text{wrd}}(w) \cdot e_{\text{ctx}}(c))}{Z(w)}$. Each word has two different embedding: e_{wrd} and e_{ctx}, either throw away e_{ctx} or concatenate them. Estimated via MLE.

Evaluation of embeddings

1. Word similarity: compare embedding similarity of similar words.
2. Word analogies: king - queen = man - woman.

Pipeline of Sentiment Analysis

1. Preprocess: tokenization, stemming, stop word removal, etc.
2. Word embedding using some models.
3. Sentence embedding by pooling word embeddings: sum, mean, max, etc.
4. Run MLP on the sentence embedding which has a fixed size.

4 Language Modeling with n-gram & RNN

Goal

Model the distribution over all finite sequences from V^*, where V is the vocabulary set. The instance space is infinite.

Local Normalization

Def: the sum of the probability of all children given their parents is 1. This guarantees the normalizer is 1.

Necessary Condition: every node has "EOS" as descendant (could be child or grandchild, etc.), o.w. there is a possible sequence of infinite length.

Sufficient Condition: every node x has "EOS" as a child and $P(\text{EOS} \mid \text{x}) \geq c$ for some positive constant c.

n-gram Assumption

n-gram simplifies the problem by assuming each word only depends on the previous $n-1$ words, i.e., $p(y_t \mid y_{<t}) := p(y_t \mid y_{t-(n-1):t-1}) := \frac{\exp(w_{y_t} \cdot h_t)}{\sum_{y'} \exp(w_{y'} \cdot h_t)}$, where $w_y \in R^d$ is the word vector and h_t is the n-gram context vector which encodes the previous $n-1$ words. "BOS" is padded for some $t < n$. $h_t := f(y_{t-1}, \ldots, y_{t-(n-1)})$ is computed by a neural network.

Recurrent Neural Network

Change $h_t := f(h_{t-1}, y_{t-1})$, where f is still a neural network. This allows information from arbitrary distance.

1. Vanilla RNN: $h_t = \sigma(W_1 h_{t-1} + W_2 e(y_{t-1}))$, $\sigma = \tanh$.
2. LSTM: has a short-term memory h_t and long-term memory c_t.
3. GRU: a fix to the vanishing/exploding gradients.

5 Part-of-Speech Tagging with CRF

Goal

Classify the grammatical categories of each word, e.g. noun, verb, etc. $p(t \mid w) = \frac{\exp(\text{score}(t,w))}{Z(w)}$ for a tagging sequence t and a sentence w of length N.

Conditional Random Field

Idea: use additively decomposable score function. $\text{score}(w,t) = \sum_{n=1}^{N} \text{score}(< t_{n-1}, t_n >, w)$.

The normalizer can be computed by DP (semiring: $(R^+ \cup \{+\infty\}, +, \times, 0, 1)$):

$$\beta(\mathbf{w}, t_N) = \mathbf{1}$$
$$\textbf{for } n = N-1, \ldots, 0:$$
$$\quad \beta(\mathbf{w}, t_n) = \bigoplus_{t_{n+1} \in \mathcal{T}} \exp\{\text{score}\langle t_n, t_{n+1} \rangle, \mathbf{w})\} \otimes \beta(\mathbf{w}, t_{n+1})$$

and $Z(w) = \beta(w, t_0)$.

Decoding the best POS tagging

Equivalent to compute the maximum-score path. Use Viterbi (semiring: $([0,1], \max, \times, 0, 1)$).

Semiring

Def: (1) addition should be commutative, (2) multiplication should distribute over addition, (3) addition and multiplication have identities, and (4) identity of addition should be an annihilator for multiplication.

Semiring	Set	$\oplus$	$\otimes$	$\bar{0}$	$\bar{1}$	intuition/application
Boolean	$\{0,1\}$	$\vee$	$\wedge$	0	1	logical deduction, recognition
Viterbi	$[0,1]$	max	$\times$	0	1	prob. of the best derivation
Inside	$\mathbb{R}^+ \cup \{+\infty\}$	$+$	$\times$	0	1	prob. of a string
Real	$\mathbb{R} \cup \{+\infty\}$	min	$+$	$+\infty$	0	shortest-distance
Tropical	$\mathbb{R}^+ \cup \{+\infty\}$	min	$+$	$+\infty$	0	with non-negative weights
Counting	$\mathbb{N}$	$+$	$\times$	0	1	number of paths

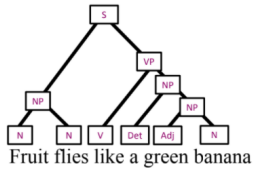
Structured Perceptron

The MLE loss for CRF is a softmax, so we add a temperature variable and take it to infinity, the loss becomes $\sum_i (\text{score}(t^i, w^i) - \max_{t'} \text{score}(t', w^i))$. This is called structured perceptron.

6 Context-Free Parsing with CKY

Goal

Find the syntax parsing tree of a context-free grammar. A constituent is a word or a group of words that function as a single unit within a hierarchical structure. Every node in the parsing tree is a constituent.



Fruit flies like a green banana

Context-Free Grammar

Components: (1) Non-terminal symbol set, (2) a start non-terminal, (3) a set of terminal symbols and (4) a set of production rules.

A grammar is in Chomsky normal form if all production rules are of the form: $N_1 \to N_2 N_3$, $N \to a$.

Probabilistic CFG

Each production rule has a probability. The probability of a tree is computed by multiplying the probabilities of all the edges. PCFG is locally normalized, making the normalizer 1.

Weighted CFG

Each production rule has a generic non-negative weight. The weights of trees are softmaxed to be a valid distribution. WCFG is globally normalized. In general, the output space is infinite, making computation of normalizer hard. However, we can efficiently compute the normalizer for a fixed sentence.

Parsing a string

$$p(t \mid s) = \frac{\exp(\text{score}(t))}{Z(s)}$$

To avoid divergence of $Z(s)$ which could be caused by the infinite number of parsing trees, we can only look at trees in CNF because the CNF theorem says that for any grammar G, there is another grammar G' that accepts the same set of strings and is in CNF. The number of possible CNF parsing tree (each non-terminal symbol has two children and each terminal symbol has one leaf child) is $O(4^N)$, no longer infinite.

CKY Algorithm

This algorithm finds all valid parsing trees under the given production rule set for CFG in CNF form in $O(N^3|R|)$, where N is the length of the string and R is the rule set.

$$\textbf{WeightedCKY}(\mathbf{s}, \langle \mathcal{N}, S, \Sigma, \mathcal{R} \rangle, \text{score}):$$
$$N \leftarrow |\mathbf{s}|$$
$$\text{chart} \leftarrow 0$$
$$\textbf{for } n = 1, \ldots, N:$$
$$\quad \textbf{for } X \to s_n \in \mathcal{R}:$$
$$\quad\quad \text{chart}[n, n+1, X] \mathrel{+}= \exp\{\text{score}(X \to s_n)\}$$
$$\textbf{for } span = 2, \ldots, N:$$
$$\quad \textbf{for } i = 1, \ldots, N - span + 1:$$
$$\quad\quad k \leftarrow i + span$$
$$\quad\quad \textbf{for } j = i+1, \ldots, k-1:$$
$$\quad\quad\quad \textbf{for } X \to Y\,Z \in \mathcal{R}:$$
$$\quad\quad\quad\quad \text{chart}[i,k,X] \mathrel{+}= \exp\{\text{score}(X \to Y\,Z)\} \times \text{chart}[i,j,Y] \times \text{chart}[j,k,Z]$$
$$\textbf{return } \text{chart}[1, N+1, S]$$

Using the $(\max, \times)$ semiring decodes the best parsing.

7 Dependency Parsing using MTT

Goal

Construct dependency tree (DT) with syntactic relations, e.g., noun-modifier, determiner, etc. The additional root node should only have degree 1.

Relation to context free grammar

CRF has no information about syntactic relation, DT has no information about constituency structure.

Types

(1) Projective DT: no crossing arcs. Algorithms are generally dynamic programming. (2) Non-Projective DT: crossing arcs. Algorithms use matrix-tree theorem (MTT).

Edge-Factored Assumption targeted the difficulty

Assume $\text{score}(t, w)$ is the product of edge scores (including the root edge).
Define $A_{ij} = \exp(\text{score}(i, j, w))$ be the edge score, $\rho_j = \exp(\text{score}(j, w))$ be the root score. Then MTT says $Z = |L|$ for following L which is $O(n^3)$:

$$L_{ij} = \begin{cases} \rho_j & \text{if } i = 1 \\ \sum_{i'=1, i' \neq j}^{n} A_{i'j} & \text{if } i = j \\ -A_{ij} & \text{otherwise} \end{cases}$$

Decoding the best DT

Equivalent to find the best directed spanning tree starting from root and the degree of root is 1. Greedy algorithms that work in undirected graph do not work.
We apply Chu-Liu-Edmonds Algorithm which is $O(n^3)$: (1) Find the best *incoming* edge for each vertex. (2) Contract cycles to be a single node c and reweight the *incoming* edge to c by adding the valid weights in c if the edge is chosen. (3) The graph now has a spanning tree. If the root constraint is not satisfied, reweight each *outcoming* edge from root to v by subtracting the weight of next best *incoming* edge to v. Remove the lowest *outcoming* edge from root and repeat step 3. (4) Expand contract nodes by breaking cycles accordingly.

8 Semantic Parsing

Goal

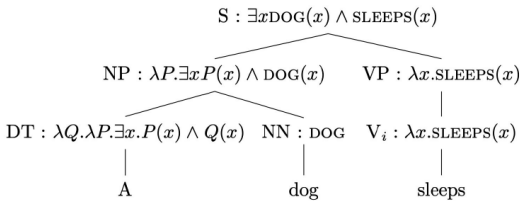
Parse of meaning. Represented by logical form composed of variables, predicates, quantifiers and boolean. Example: $\forall p.(Person(p) \rightarrow \exists q.(Person(q) \land p \neq q \land \text{Loves}(p, q)))$.

Principle of Compositionality

The meaning of a complex expression is a function of the meanings of that expression's constituent parts.

Enriched lambda calculus to represent meanings

(1) Logical constants such as entities (e.g., ALEX) and relations (e.g., likes). (2) Variables, which are undetermined objects, similar to free variables in lambda calculus. (3) Literals such as LIKES(ALEX, x), formed by applying relations to objects. (4) Logical terms built using literals with logical connectives (e.g., $\land$) and quantifiers (e.g., $\exists$). (5) lambda terms built using lambda operator.



9 Transliteration with WFST

Goal

Transliteration is to "spelling out" a word in another alphabet, i.e., replace the characters with phonetic approximations, by mapping strings in one set to another set.
Weighted finite-state transducers is a probabilistic model that map strings from input vocabulary to output vocabulary. The number of states of our modeled language is finite, and the transition is weighted. Each transition is labeled by an input character, an output character and a weight. If the transition does not include the output character, then it is a weighted finite-state acceptor.

The probabilistic model for transition sequences

Let π be a path that generates the input sequence X and output sequence Y. By definition, $\text{score}(\pi) = \sum_i \text{score}(\pi_i)$. Therefore, $p(y \mid x) = \frac{1}{Z} \sum_\pi \exp(\sum_i \text{score}(\pi_i))$, where Z is a sum over the whole output space which is infinite.

Floyd-Warshall Algorithm to compute the shortest path for all pairs in the graph without negative cycles

```
let dist be a |V| × |V| array of minimum distances initialized to ∞ (infinity)
for each edge (u, v) do
    dist[u][v] ← w(u, v)  // The weight of the edge (u, v)
for each vertex v do
    dist[v][v] ← 0
for k from 1 to |V|
    for i from 1 to |V|
        for j from 1 to |V|
            if dist[i][j] > dist[i][k] + dist[k][j]
                dist[i][j] ← dist[i][k] + dist[k][j]
            end if
```

$dist^k$ naturally encodes the shortest path between two nodes of maximum length k.

Normalizer computation

Let α be the vector of weights of starting in the states and β be the vector of weights of ending in the states. Let W^ω be a matrix s.t. W^ω_{nm} is the weight from n to m with arc labeled by ω, ω is an element of output space. Then $Z = \alpha^T (\sum_{\omega \in \Omega \cup \{\epsilon\}} W^\omega)^* \beta$, where the $*$ (Kleene closure) is computed by Floyd-Warshall using semiring $(R^+ \cup \{+\infty\}, \min, +, +\infty, 0)$.

10 Machine Translation with Transformer

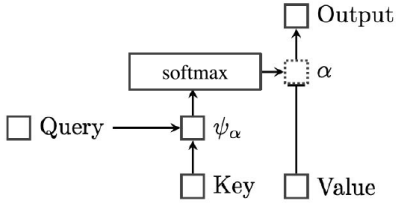
Goal

Translate a sentence from a source language to the target language. The problem is that there are many correct translations.

Seq2Seq Model

Basically a representation method: $z = \text{encoder}(x)$ and $y \mid x \sim \text{decoder}(z)$. We iteratively model $p(y_t \mid x, y_1, \ldots, y_{t-1})$ because $p(y \mid x) = \prod_t p(y_t \mid x, y_1, \ldots, y_{t-1})$.
At inference time, the model first predicts y_1 by $p(y_1 \mid x)$, then y_2 by $p(y_2 \mid x, y_1)$ and more.
For standard RNN, the decoder only receives one vector, causing information bottleneck.

Attention Mechanism



Standard attention: what information from encoder is relevant for decoding step t. Use the output of encoder as key and value and the output of decoder at current step as the query.
Self-attention: what information from inputs are relevant for encoding step t or what information from (previous) outputs are relevant for decoding step t. Use only encoder's or decoder's output as key, query and value.

11 Bias and Fairness

Goal

Consider bias in the labeling, sample selection, task, features, loss functions and feedback loops.

Train-Test Mismatch

Problem: distribution mismatch between old and new data.

Unsupervised Adaption (no label of new data)

Importance sampling: $\mathbb{E}_{(x,y)\sim\mathcal{D}^{\text{new}}} l(y, f(x)) = \mathbb{E}_{(x,y)\sim\mathcal{D}^{\text{old}}} (\frac{\mathcal{D}^{\text{new}}(x,y)}{\mathcal{D}^{\text{old}}(x,y)} l(y, f(x)))$. Since $\frac{\mathcal{D}^{\text{new}}(x,y)}{\mathcal{D}^{\text{old}}(x,y)}$ is unknown, we could use approximations: $\mathcal{D}^{\text{new}} \propto \mathcal{D}^{\text{base}} p(s = 0 \mid x)$ and $\mathcal{D}^{\text{old}} \propto \mathcal{D}^{\text{base}} p(s = 1 \mid x)$. Therefore, $\frac{\mathcal{D}^{\text{new}}(x,y)}{\mathcal{D}^{\text{old}}(x,y)} = \frac{1}{p(s=1|x)} - 1$. We can train a classifier to distinguish between old and new distribution and thus get the estimation of $p(s = 1 \mid x)$.

Supervised Adaption (some labels of new data)

Use feature augmentation (separation) by forcing features into three components: shared, old-only (forced to be zero in the new dataset) and new-only (forced to be zero in the old dataset).