1 Dependency Parsing using MTT

Definition

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 $\mathrm{score}(t, w)$ is the product of edge scores (including the root edge).

Define $A_{ij} = \exp(\mathrm{score}(i, j, w))$ be the edge score, $\rho_j = \exp(\mathrm{score}(j, w))$ 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.

2 Semantic Parsing

Definition

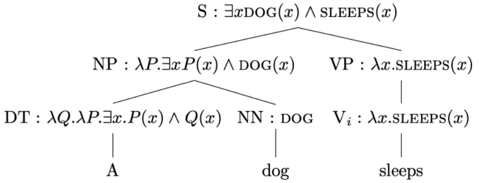
Parse of meaning. Represented by logical form composed of variables, predicates, quantifiers and boolean. Example: $\forall p.(Person(p) \rightarrow \exists q.(Person(q) \wedge p \neq q \wedge \mathrm{Loves}(p, q)))$.

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., $\wedge$) and quantifiers (e.g., $\exists$). (5) lambda terms built using lambda operator.



3 Transliteration with WFST

Definition

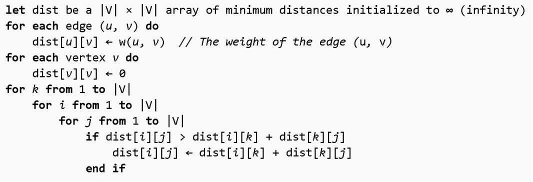
Transliteration is to "spelling out" a word in another alphabet, i.e., replace the characters with phonetic approximations, by mapping strings in one set to another set.

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, $\mathrm{score}(\pi) = \sum_i \mathrm{score}(\pi_i)$. Therefore, $p(y \mid x) = \frac{1}{Z} \sum_\pi \exp(\sum_i \mathrm{score}(\pi_i))$, 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 $(R^+ \cup \{+\infty\}, \min, +, +\infty, 0)$.

4 Machine translation with transformer

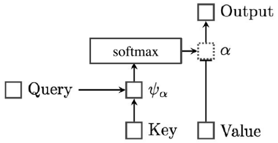
Seq2Seq Model

Basically a representation method: $z = \mathrm{encoder}(x)$ and $y \mid x \sim \mathrm{decoder}(z)$. 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.