

TREE-STRUCTURED DECODING WITH DOUBLYRECURRENT NEURAL NETWORKS

Шокоров Вячеслав Александрович

Московский физико-технический институт
Факультет управления и прикладной математики
Кафедра интеллектуальных систем

Москва,
2022 г.

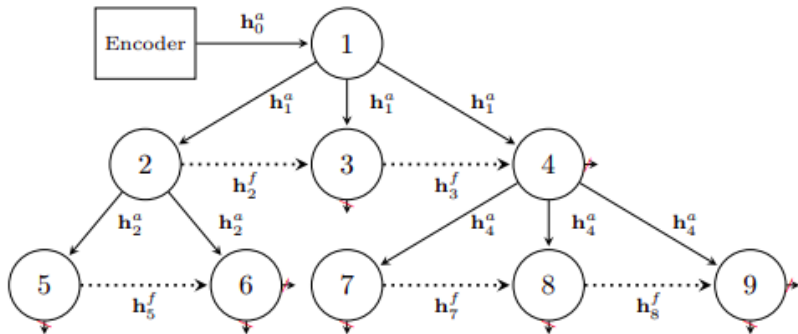
Кратко

В работе предлагается архитектура нейронной сети для генерации объектов структурированных в виде дерева из векторных представлений. Ключевым моментом является применение дважды рекуррентной модели нейронной сети, состоящей из отдельных повторений ширины и глубины, которые объединяются внутри каждой ячейки (узла) для генерации выходных данных.

Проблема

Объекты структурированные в виде дерева не имеют регулярную структуру.

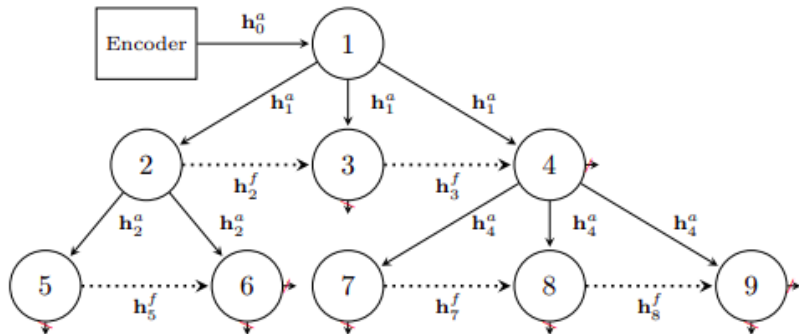
Схема применения DRNN



Узлы помечены в том порядке, в котором они были сгенерированы. Сплошные (пунктирные) линии указывают на связи идущие от родителя (соседа). Скрещенные стрелки указывают на остановку генерации листьев на данном уровне.

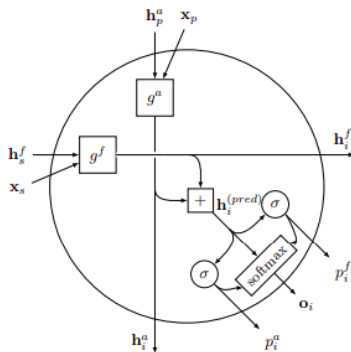
- Авторы предложили новую архитектуру нейронной сети, специально разработанную для декодирования с древовидной структурой, которая поддерживает отдельные повторяющиеся состояния глубины и ширины и объединяет их для получения скрытых состояний для каждого узла в дереве.
- Они оснастили эту новую архитектуру механизмом для явного прогнозирования топологии дерева (в отличие от неявного добавления узлов со специальными токенами).
- Также экспериментально показывали, что предлагаемый метод способен восстанавливать деревья из закодированных представлений и что он превосходит современные методы в задаче, состоящей из сопоставления предложений с простыми функциональными программами.

Схема применения DRNN



$$h_i^a = g^a(h_{p(i)}^a, x_{p(i)}) \quad h_i^f = g^f(h_{s(i)}^f, x_{s(i)})$$
$$h_i^{(pred)} = \tanh(U^f h_i^f + U^a h_i^a)$$

Схема применения DRNN



$$p_i^a = \sigma(u^a \cdot h_i^{(pred)}) \quad p_i^f = \sigma(u^f \cdot h_i^{(pred)})$$

$$o_i = \text{softmax}(Wh_i^{(pred)} + \alpha_i v^a + \varphi_i v^f) \quad \alpha_i, \varphi_i \in \{0, 1\}$$

Во время обучения $\alpha_i = 1$, если у узла i есть дочерние узлы, и $\varphi_i = 1$, если у него есть соседний. Во время тестирования эти значения получаются из биномиального распределения с вероятностями p_i^a, p_i^f соответственно.

$$L(\hat{x}) = \sum_i L^{label}(x_i, \hat{x}_i) + L^{topo}(p_i, \hat{p}_i)$$

- Задача построения векторного представления дерева.
- Задача получения программного кода/псевдокода по тексту.
- Задача машинного перевода.