1 Tensor Networks

Our goal now is to focus on a subclass of models over Σ^*. To this end, we analyze *tensor networks*.

We notate a tensor T_v with k axes of sizes $D_v = (d_1, \dots, d_k)$ as $T_v : [d_1] \times \cdots \times [d_k] \mapsto \mathbb{R}$. Because indexing is mostly clear based on the context, we treat D_v as a multiset. Thus, when contracting two tensors T_u and T_v over a shared axis of size d_e, the output tensor T_C reads as:

$$T_C : (D_v \setminus \{d_e\}) \cup (D_w \setminus \{d_e\}) \mapsto \mathbb{R},\ i \mapsto \sum_{i_e \in [d_e]} T_u(i) T_v(i) \quad .$$

Definition 1.1 (Tensor Network over Σ^n)**.** A *tensor network* $\mathcal{T}$ over Σ^n is a graph $G = (V, E)$, where:

- V is a set of vertices, where each vertex $v = (\text{layer}, \text{index}) \in V$ represents a tensor T_v of dimensions $D_v = \{d_1, \dots, d_k\}$. The indexing is implicitly defined by the context. V_{layer} notates all vertices in the specified layer.
- I is a set of input tensors $I = (T_{0,1}, \dots, T_{0,n})$, where every tensor has exactly one axis of size $|\Sigma|$.
- E is a set of edges, where each edge $e = \{u, v\} \in E,\ u \in V_l,\ v \in V_{l+1}$ represents a shared index of dimension size d_e between two tensors T_u and T_v, which is summed over in the contraction process.
- For every node v, $\deg(v)$ must match the number of axes $|D_v|$ of v.
- Then, when initializing the input tensors by an one-hot encoding representing the presence of the specified token at that location, $\mathcal{T}$ assigns a probability to $w \in \Sigma^n$ by:

  $$S_{n,\mathcal{T}}(w) \coloneqq \frac{\mathcal{T}(w)}{\sum_{w' \in \Sigma^n} \mathcal{T}(w')} \quad ,$$

  where $\mathcal{T}(w)$ is the scalar output of the network upon contracting it with an initialization specified by w.

Remark 1.1. We might expand the definition by allowing multi-edges, i.e. contracting more than two tensors over a common index at once.