1 Tensor Networks

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

We denote a tensor T_v with k axes of sizes $D_v = \{d_1, \ldots, d_k\}$ as a function

$$T_v : [d_1] \times \cdots \times [d_k] \mapsto \mathbb{R} \quad .$$

As a shorthand, we write

$$[D_v] \coloneqq [d_1] \times \cdots \times [d_k] \quad .$$

Since indexing is usually clear from context, we treat D_v as a multiset of axis sizes.

Given two tensors T_u and T_v that share a common axis of size d_e, their contraction over this axis produces a new tensor T_C with dimension set

$$D_C = (D_u \setminus \{d_e\}) \cup (D_v \setminus \{d_e\}) \quad ,$$

defined as

$$T_C(i) = \sum_{i_e \in [d_e]} T_u(i_{D_u}, i_e) \cdot T_v(i_{D_v}, i_e) \quad ,$$

where $i \in [D_C]$. Note that $d_e \notin D_C$, which is why we explicitly included index i_e in the summation.

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

- V is the set of vertices, where each vertex $v = (\text{layer}, \text{index}) \in V$ corresponds to a tensor T_v with axis sizes $D_v = \{d_1, \ldots, d_k\}$. Let $V_{\text{layer}} \subseteq V$ denote the set of all vertices at a given layer.
- The input set $I = (T_{0,1}, \ldots, T_{0,n}) \subset V$ consists of tensors each having a single axis of size $|\Sigma|$. These serve as the one-hot-encoded inputs corresponding to a string $w \in \Sigma^n$.
- $E \subseteq \{\{u, v\} \mid u \in V_l,\, v \in V_{l+1}\}$ is the set of edges. Each edge $e = \{u, v\}$ represents a shared index of size d_e between tensors T_u and T_v, which is summed over during contraction.
- The usual tensor network constrains: For each vertex $v \in V$, the degree $\deg(v)$ must match the number of axes $|D_v|$, and shared indices must correspond to same axis sizes.

Once the input tensors are initialized with one-hot encodings derived from a string $w \in \Sigma^n$, the network computes a scalar output $\mathcal{T}(w)$. This induces a probability distribution over Σ^n defined by:

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

where $f : \mathbb{R} \mapsto \mathbb{R}_{\geq 0}$ is any arbitrary function like $f \equiv \exp$.

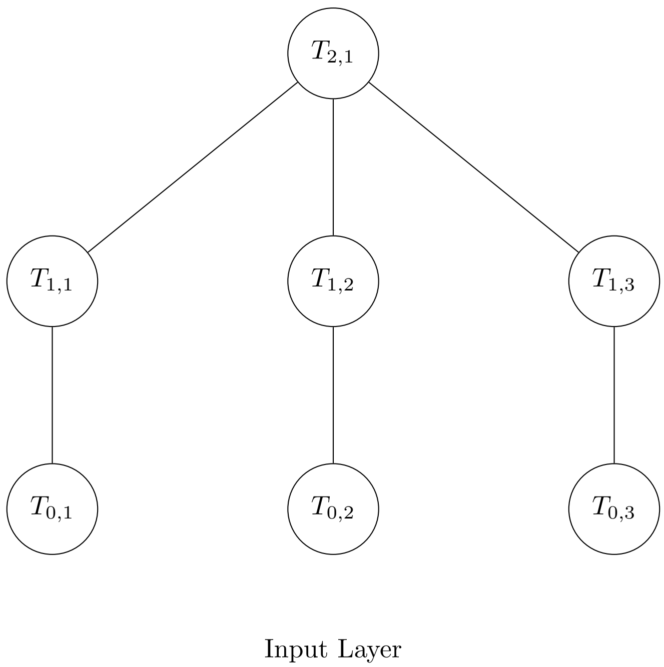

Figure 1: A basic tensor network over Σ^3.

Definition 1.2 (Normalized and Non-Negative Tensor Networks)**.** Let $\mathcal{T}$ be a tensor network over Σ^n with scalar output $\mathcal{T}(w)$ for each $w \in \Sigma^n$. Define the total mass of the network as

$$|\mathcal{T}| := \sum_{w \in \Sigma^n} \mathcal{T}(w) \quad .$$

We say $\mathcal{T}$ is *normalized* iff $|\mathcal{T}| = 1$.

Furthermore, a tensor network is said to be *non-negative* iff for all $w \in \Sigma^n$ we have $\mathcal{T}(w) \geq 0$.

Remark 1.1. We can enforce all tensor networks of our model space to be non-negative by only allowing for non-negative tensors in the networks.

Definition 1.3 (Normalization of Tensor Networks)**.** Let $\mathcal{T}$ be a tensor network over Σ^n, and let $H \coloneqq V \setminus I$ be the set of non-input tensors, and define $|H|$ as its cardinality. The *induced normalized tensor network* $\dfrac{\mathcal{T}}{|\mathcal{T}|}$ is the same network as $\mathcal{T}$, but each entry of each tensor in H is scaled by the factor $\dfrac{1}{\sqrt[|H|]{|\mathcal{T}|}}$.

Lemma 1.1. *Let $J \subseteq [n]$ and let $\mathcal{T}$ be a tensor network over Σ^n. Define a modified network $\mathcal{T}_J$ where for all $j \in J$, the input tensor $T_{0,j}$ is initialized to the all-ones vector (i.e., $\mathbf{1} \in \mathbb{R}^{|\Sigma|}$). Then for any $w \in \Sigma^{[n]\setminus J}$:*

$$\sum_{w_J \in \Sigma^{|J|}} \mathcal{T}(w_J, w) = \mathcal{T}_J(w) \quad .$$

Proof. We proceed by induction on the size of the subset $J \subseteq [n]$.

Base case: $|J| = 0$. Then $J = \emptyset$, so $\mathcal{T}_J = \mathcal{T}$, and the sum over $w_J \in \Sigma^{|J|}$ is a sum over a singleton (the empty word), yielding:

$$\sum_{w_J \in \Sigma^0} \mathcal{T}(w_J, w) = \mathcal{T}(w),$$

and since $\mathcal{T}_J(w) = \mathcal{T}(w)$, the base case holds.

Inductive step: Assume the lemma holds for all subsets of size k, and let $J \subseteq [n]$ with $|J| = k + 1$. Pick any $j_0 \in J$, and define $J' = J \setminus \{j_0\}$, which has size k. By the inductive hypothesis, for any $w \in \Sigma^{[n]\setminus J}$, we have:

$$\sum_{w_{J'} \in \Sigma^k} \mathcal{T}(w_{J'}, w, w_{j_0}) = \mathcal{T}_{J'}(w, w_{j_0}) \quad ,$$

where $w_{j_0} \in \Sigma$ varies over its values.

Now consider the sum over all $w_J \in \Sigma^{k+1}$, which we write as:

$$\sum_{w_{J'} \in \Sigma^k} \sum_{w_{j_0} \in \Sigma} \mathcal{T}(w_{J'}, w_{j_0}, w) \quad .$$

By the inductive hypothesis, this equals:

$$\sum_{w_{j_0} \in \Sigma} \mathcal{T}_{J'}(w_{j_0}, w) \quad .$$

Observe that in $\mathcal{T}_{J'}$, the input tensor at position j_0 is still initialized to a one-hot vector, while the inputs at J' have been replaced with the all-ones vector.

Now, note that the inner sum over w_{j_0} is equivalent to replacing the input at j_0 with the all-ones vector, since the sum represents a sum over vector dot products of vector $\boldsymbol{v}_{w_{j_0}} := \mathcal{T}_{J'}(w_{j_0}, w)$ with one-hot encoded vectors. It follows from linearity that we can factor out $\boldsymbol{v}$, and the sum of the one-hot encoded vector yields the all-ones vector. Thus:

$$\sum_{w_{j_0} \in \Sigma} \mathcal{T}_{J'}(w_{j_0}, w) = \mathcal{T}_J(w) \quad .$$

Hence, by induction, the lemma holds for all subsets $J \subseteq [n]$.

□

Corollary 1.1. *Let $\mathcal{T}$ be a tensor network over Σ^n, and let $\mathcal{T}_{[n]}$ be the network where all input tensors are initialized to the all-ones vector. Then:*

$$\mathcal{T} \text{ is normalized} \iff \mathcal{T}_{[n]} = 1 \quad ,$$

i.e., the total contraction of the network with all-one input tensors equals 1.

Lemma 1.2. *Let $\mathcal{T}$ be a tensor network over Σ^n. The induced normalized tensor network $\frac{\mathcal{T}}{|\mathcal{T}|}$ is indeed normalized, and if additionally $\mathcal{T}$ is non-negative and $f \equiv id$, we have for all $w \in \Sigma^n$:*

$$S_{n,\mathcal{T}}(w) = S_{n,\frac{\mathcal{T}}{|\mathcal{T}|}}(w) \quad .$$

Proof. Let H be the set of non-input tensors in $\mathcal{T}$, and let $|H| = m$. In the induced normalized network, every tensor in H is scaled by a factor $\alpha = \dfrac{1}{\sqrt[m]{|\mathcal{T}|}}$. Since the final output $\mathcal{T}(w)$ is a multilinear contraction over the tensors, this means the scalar output for any $w \in \Sigma^n$ becomes:

$$\left(\prod_{v \in H} \alpha\right) \cdot \mathcal{T}(w) = \alpha^m \cdot \mathcal{T}(w) = \frac{1}{|\mathcal{T}|} \cdot \mathcal{T}(w) \quad .$$

Hence,

$$\left(\frac{\mathcal{T}}{|\mathcal{T}|}\right)(w) = \frac{\mathcal{T}(w)}{|\mathcal{T}|} \quad .$$

Summing over all $w \in \Sigma^n$,

$$\left|\frac{\mathcal{T}}{|\mathcal{T}|}\right| = \sum_{w \in \Sigma^n} \frac{\mathcal{T}(w)}{|\mathcal{T}|} = \frac{1}{|\mathcal{T}|} \sum_{w \in \Sigma^n} \mathcal{T}(w) = \frac{|\mathcal{T}|}{|\mathcal{T}|} = 1 \quad .$$

Moreover, since the normalization rescales all outputs by the same constant, the ratio of the terms to the total sum remains unchanged:

$$S_{n,\frac{\mathcal{T}}{|\mathcal{T}|}}(w) = \frac{\left(\frac{\mathcal{T}(w)}{|\mathcal{T}|}\right)}{\sum_{w' \in \Sigma^n} \left(\frac{\mathcal{T}(w')}{|\mathcal{T}|}\right)} = \frac{\mathcal{T}(w)}{|\mathcal{T}|} \cdot \frac{1}{1} = S_{n,\mathcal{T}}(w) \quad .$$

This completes the proof. ☐

One might ask whether our definition for tensor networks is bit restrictive, as it only allows for contraction over *pairs* of tensors. But what if we wanted to contract, say, three tensors at once over a common index?

Proposition 1.1. *Let $V' \subseteq V$ be a set of tensors in a tensor network, each containing an axis of dimension d labeled by a shared index i. Contracting all tensors in V' over the shared index i is equivalent to contracting each tensor individually with a single tensor*

$$\delta_{|V'|} : [d]^{|V'|} \mapsto \mathbb{R}_{\geq 0}$$

defined by

$$\delta_{|V'|}(i_1, \ldots, i_{|V'|}) = \begin{cases} 1 & \text{if } i_1 = \cdots = i_{|V'|} \ , \\ 0 & \text{otherwise.} \end{cases} \quad .$$

That is, a full contraction over a shared index can be implemented by introducing a single copy tensor connected to each tensor in V'.

Proof. Each tensor T_v for $v \in V'$ has an index $i \in [d]$ corresponding to the shared axis. The contraction over this index is defined by summing over the common value of i across all tensors:

$$\sum_{i=1}^{d} \prod_{v \in V'} T_v(\ldots, i, \ldots) \quad .$$

Now consider a new tensor $\delta_{|V'|}$ of order $|V'|$, defined as 1 if all indices are equal and 0 otherwise. Let each tensor T_v maintain its original indices, but connect to $\delta_{|V'|}$ via the position corresponding to v.

The contraction over this shared structure gives:

$$\sum_{i_1, \ldots, i_{|V'|}} \left(\prod_{v \in V'} T_v(\ldots, i_v, \ldots) \right) \delta_{|V'|}(i_1, \ldots, i_{|V'|}) \quad .$$

By definition of $\delta_{|V'|}$, this enforces $i_1 = \cdots = i_{|V'|}$, reducing the above to:

$$\sum_{i=1}^{d} \prod_{v \in V'} T_v(\ldots, i, \ldots) \quad ,$$

which is exactly the original contraction. Hence, the two constructions are equivalent. $\square$

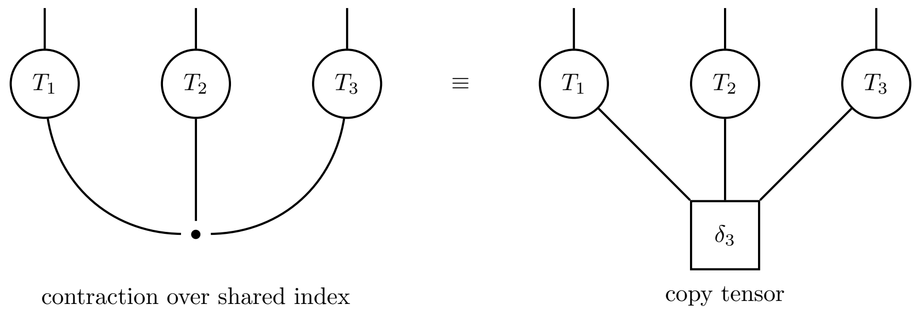

Figure 2: Contracting multiple tensors over one shared index is equivalent to contracting them individually with a single copy tensor.

1.1 Bulk Marginal Property

We are interested in tensor networks that have the bulk marginal property. When further specifying our network structure, we might have a model space for varying word lengths n, but not for every $n \in \mathbb{N}$. Take for example the model space of binary trees as shown in figure 3.

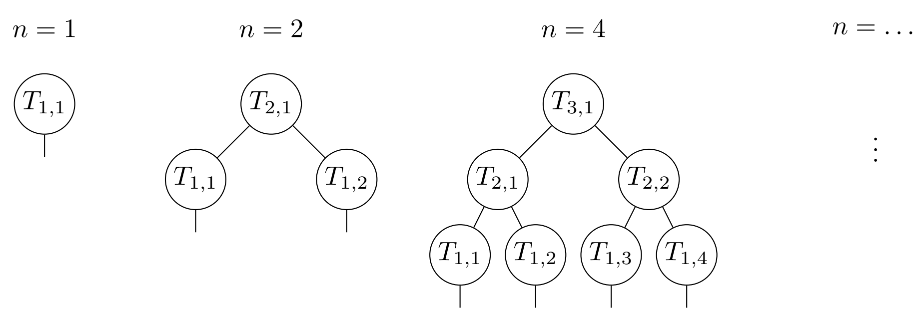

Figure 3: A model space for sequences of length $n = 2^k$.

In definition **??** we saw how to construct a model with the desired bulk marginal property based on the base model. However, we might not always have a base model for ever $n \in \mathbb{N}$ like discussed. Luckily, it turns out that this is not an issue, as there are many ways we can build a new model with the bulk marginal property from a base model even if it is only defined on a subset of $\mathbb{N}$. Without a proof, we might do the same procedure as in definition **??** but with bigger steps (instead of taking always the consecutive model), and induce the in-between models by marginalizing the bigger ones.

Alternatively, if we wanted a model with bulk marginal property that itself is also an element of our specified model space, we might ask ourselves, how we can construct a bigger tensor network while preserving the distribution in its leading random variables.

Let's analyze the following example: Say we wanted to integrate the tree tensor network for $n = 2$ in figure 3 into a bigger tree tensor network with $n = 4$. We assume that our network is normalized. Thus, in order for our new tensor network to have the bulk marginal property, contracting the smaller network must be equivalent to contracting the bigger one, where the new nodes (in this case $T_{1,3}$ and $T_{1,4}$) are contracted with all-ones vectors.

But there is one important question: How do we actually connect the smaller network to the new nodes? Because all nodes represent tensors that have a fixed number of axes, so we cannot just draw an additional edge connecting them with the new nodes. However, we might use proposition 1.1 to contract multiple tensors over a shared index. Hence, we might use the following structure as depicted in figure 4.

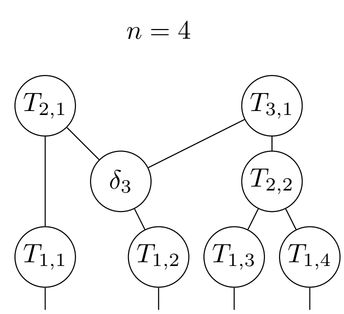

Figure 4: Integrating a smaller tensor network into a bigger one based on figure 3.

Like already discussed, because we want to ensure the bulk marginal property, we must have equivalence of the following tensor networks in figure 5.

Note that if we indeed had equivalence of these models, this would imply that the bigger model is now normalized as well based on lemma 1.1.

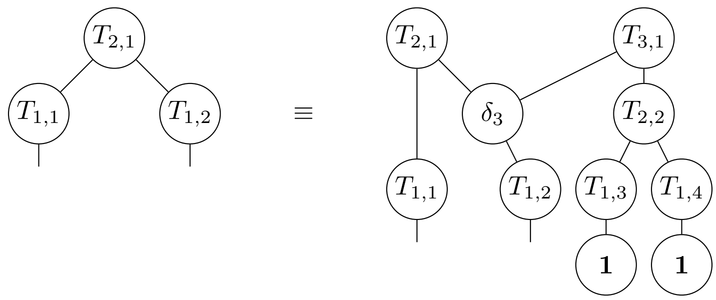

Figure 5: Bulk marginal property enforces the equivalence of these models.

We can ensure equivalence if the new tensors of the network contract to an all-ones vector like depicted in figure 6.

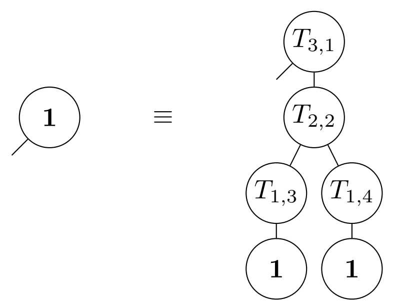

Figure 6: It is sufficient for the bulk marginal property that this tensor network evaluates to an all-ones vector.

We may call this approach of integrating a smaller tensor network into a bigger one as *structural integration*. However, there might be other options to ensure bulk marginal property: Lets assume all tensors of figure 3 for $n = 4$ have free parameters except tensor $T_{1,1}$. Then, when assigning all-ones vectors to $T_{1,3}$ and $T_{1,4}$, we get a matrix upon contracting all tensors except for $T_{1,1}$. But $T_{1,1}$ itself is a matrix, and so the probability distribution of the first two random variables is completely defined by the multiplication of these two matrices. Now, assuming that the matrix dimensions are sufficient, we can completely restore the original distribution by defining the entries of $T_{1,1}$ accordingly (assuming that it is possible).