To answer this question and to allow the network to learn the optimal location for writing and reading, we need to represent the problem by a differentiable function (so that an optimizer can be trained to learn its parameters). In fact we want to make the function differentiable with respect to the reading/writing location, so that the network can learn where to read and write. NTM does this by writing and reading anywhere in the vector, but with a different weighting for different locations.

The read result is a weighted sum.

$$r \leftarrow \sum_i a_i M_i$$

That is, rather than assigning a single location the network should write to, the NTM produces an attention distribution and a set of weights designating how much to write at each location, and the read result is the weighted sum of the written values at the different locations.

Two different techniques serve for training the network to learn the weights: (1) content-based attention and (2) location-based attention. Content-based attention enables the NTM to learn to focus on the most relevant areas of the memory vector array, while location-based attention enables more flexibility and movement throughout the memory.

Attentional Interfaces

In learning problems there is an abundance of information and in many cases much of it is not directly related or relevant to the problem at hand. The purpose of the attentional interface is to learn what part of the data is relevant. In other words, its function is to help the network learn to focus on a subpart of the input data. The attentional interface implements an approach resembling that of NTM: we focus on the entire input, but with differing weights.

Network B focuses on a different part of the data than network A, at every stage.