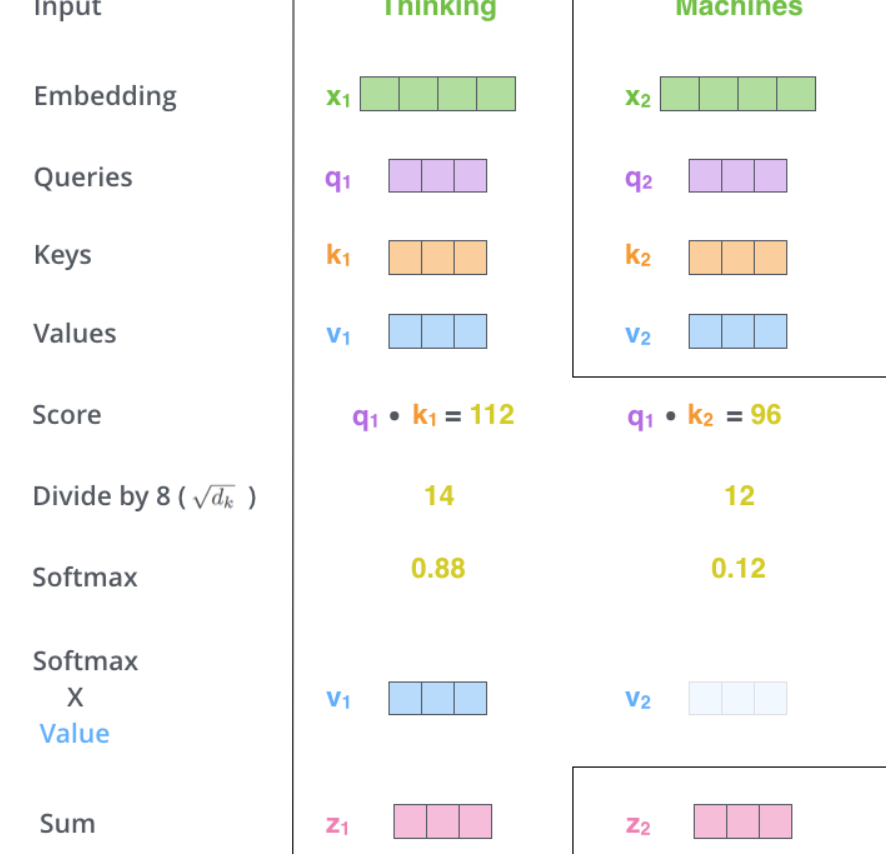
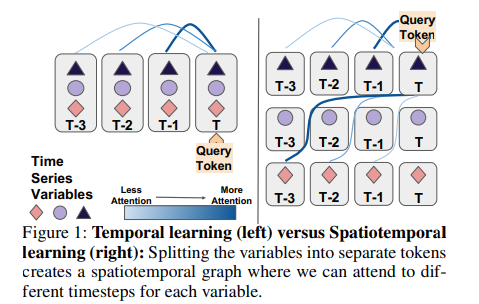
Both temporal and spatial relationships (how variables impact on one another)

* State-of-the-art forecasting models rely on neural attention between timesteps.(temporal learning)
* This allows for temporal learning but fails to consider distinct spatial relationships between variables.
* convert multivariate Time series forecasting into a novel “spatiotemporal sequence”
* each input token represents the value of a single variable at a given timestep
* Long-Range Transformers can then learn interactions between space, time, and value information jointly along this extended sequence.
* Current state-of-the art TSF models substitute classic Seq2Seq architectures for neural-attention-based mechanisms.
* flatten multivariate inputs into long sequences where each input token isolates the value of a single variable at a given timestep.
* Long-Range Transformer architectures learn self-attention networks across both space and time jointly.
* This method can interpret long context windows and forecast many timesteps into the future while also discovering the spatial relationships between hundreds of variables
* GNNs rely on predefined graphs representing the relationships between input variables
* Deep Learning approaches to TSF are generally based on a Seq2Seq framework
* a context window of the recent past is mapped to a target window of predictions for the future.
* Informer encodes time series inputs with learned temporal embeddings and appends a start token to the decoder sequence of zeros that need prediction.

LONG RANGE TRANSFORMER

* Transformers are centred around the multi-head self-attention (MSA) mechanism.
* 
* Attention learns weighted relationships between its inputs in order to pass information between tokens.
* Also leads to alleviate vanishing gradient problem
* Multi self attention is not good for long inputs
* There are variants of Multi self attention
* Approximate MSA and performer
* Performer-> approximates MSA in linear space and time with a kernel of random orthogonal features and enables the long-sequence approach

Spatiotemporal forecasting

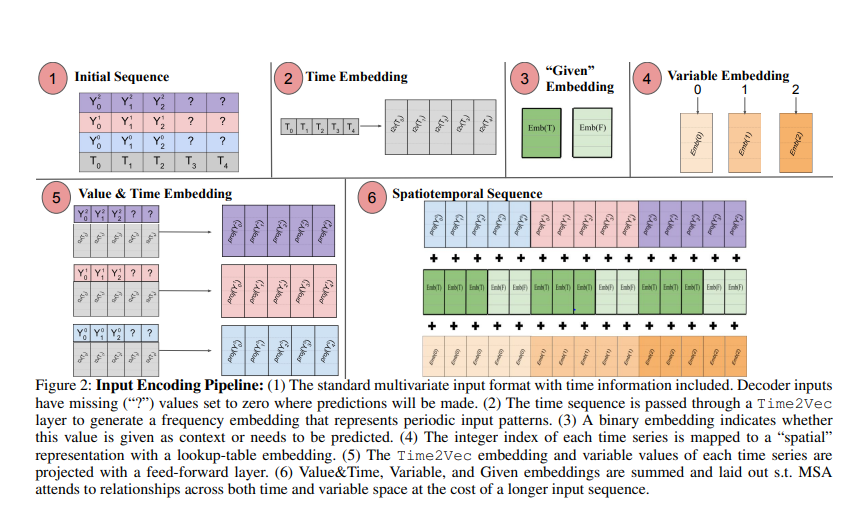
* Multivariate TSF involves two axes of complexity: the forecast sequence’s duration L, and the number of variables N
* Attention-based models need to connect the variable values at a given timestep to entire timesteps in the past.
* the variables we are modeling may have different periodicities and relations to one another.
* Example
* Suppose we are forecasting the air temperature at weather stations in the United States
* Several stations are located relatively close to each other in central Texas.
* The rest are also clustered together but centered hundreds of miles away outside New York City
* The two groups experience different weather patterns as a result of their geographic separation
* In order to predict future weather patterns in Texas, we may need to attend to the values of nearby stations when the current weather cycle began
* \*\*\*\*When faced with this problem, a temporal attention model would be forced to “average” these requirements and attend to timesteps somewhere in the middle - a compromise that provides little relevant information to either situation.
* 

METHOD

Spatiotemporal Forecasting with Transformers

* multivariate Time series forecasting inherently models multiple distinct sequences per timestep.
* our attention mechanism can scale with the additional length, it can now learn unique relationships for every variable.
* After flattening the input variables, we need to allow Transformers to correctly interpret the variable each token originates from as well as its time and scalar value
* Transformers are permutation invariant, meaning they cannot interpret the order of input tokens by default
* This is fixed by adding a position embedding to the tokens - usually with a fixed sinusoidal pattern

Input



* In a time series problem, the input sequence takes place in a specific window of time that may be relevant to long-term seasonal patterns

Example->

* it is important to know the relative order of three weeks of temperature observations, but it is also important to know that those three weeks take place in late spring when temperatures are typically rising.
* Time2Vec passes a representation of absolute time (e.g., the calendar datetime) through sinusoidal patterns of learned offsets and wavelengths.
* The concatenated timeseries value and time embedding are then projected to the input dimension of the Transformer model with a feed-forward layer. We refer to the resulting output as the “Value&Time Embedding.”
* We add a “Variable Embedding” to each token
* The variable embedding indicates the time series from which each token originates

ENCODER AND DECODER

* The encoder observes the context sequence in which all values are present. However, the decoder deals with missing values that need prediction in addition to the start token of the last few timesteps of the context sequence
* **Scaling with fast attention**
* Long sequence transformers
* ProbSparse attention (performer)

**Scaling with Initial and Intermediate Conv1Ds**. Using INFORMER

* learn to summarize the sequence with strided one-dimensional convolutions; we use the proper stride and padding to cut sequence length in half per Conv1D layer
* Initial” convolutions are applied to the Value&Time embedding while the variable embedding tokens are repeated for half their usual length
* “Intermediate” convolutions occur between MSA layers.

LOCAL AND GLOBAL ATTENTION

* learning an attention layer over a longer global sequence eliminates a helpful local bias that hurts performance in problems with large N.
* adding additional “local” attention modules to each encoder and decoder layer

output

* feed-forward layer of the decoder generates a two-dimensional vector that parameterizes the mean and standard deviation of a Normal distribution

