**SOME NOTES :**

The paper addresses a core challenge in machine translation:

* Encoder-Decoder architecture effective for short sentences this approach struggles with longer sentences because it’s difficult to capture all the necessary information in one fixed vector

To solve this, the authors propose a new model that incorporates an attention mechanism. This allows the model to focus on different parts of the input sentence as it generates each word of the translation, rather than relying on a single summary.

This mimics how humans translate, focusing on specific words or phrases as needed rather than holding the entire sentence in memory.

The Model Steps :

**Encoder**

* The encoder processes the input sentence word by word using a RNN, such as an LSTM or GRU.
* Instead of producing a single fixed vector, it generates a **sequence of hidden states**, one for each input word. These hidden states capture the context around each word in the sentence.

#### ****Decoder with Attention****

* The decoder, also an RNN, generates the translation one word at a time.
* At each step, the attention mechanism calculates a context vector, weighted sum of the encoder’s hidden states. This vector represents the most relevant parts of the input sentence for predicting the next word.
* The attention weights are computed dynamically based on the decoder’s current state and the encoder’s hidden states, allowing the model to "attend" to different input words as needed.

**Attention Mechanism**

* Mathematically, the attention weights are often computed using a small neural network or a dot-product operation, ensuring the model learns the best alignment for each translation step.

#### ****Output Prediction****

* The decoder combines the context vector with its own hidden state to predict the next word in the translation, repeating this process until the full translation is generated.

Paper link : <https://arxiv.org/abs/1409.0473>