**Neural Machine Translation (NMT)** is a recently proposed approach to machine translation that aims to build a single neural network jointly tuned to maximize translation performance, unlike traditional statistical machine translation systems which consist of many separately tuned sub-components. NMT models typically belong to a family of encoder–decoders. In this standard approach, an encoder neural network reads a source sentence and encodes it into a fixed-length vector. A decoder then generates a translation from this encoded vector. The entire encoder–decoder system is jointly trained to maximize the probability of a correct translation given a source sentence.

A significant potential issue with this basic encoder–decoder approach is that the neural network must compress all necessary information of a source sentence into a single fixed-length vector. This limitation can make it difficult for the network to handle long sentences, especially those longer than the training data sentences. Empirical studies have shown that the performance of a basic encoder-decoder rapidly deteriorates as input sentence length increases.

To address this problem, a novel architecture called **RNNsearch** is proposed. The key distinguishing feature of RNNsearch is that it **does not attempt to encode the entire input sentence into a single fixed-length vector**. Instead, it encodes the input sentence into a sequence of vectors (annotations) and adaptively chooses a subset of these vectors while decoding the translation. This allows the model to better cope with long sentences by freeing it from the requirement to squash all source sentence information into a fixed-length vector, regardless of length. The information can be spread across the sequence of annotations and selectively retrieved by the decoder.

The proposed architecture consists of a bidirectional RNN (BiRNN) as an encoder and a decoder that simulates searching through the source sentence during decoding. The BiRNN encoder reads the input sequence in both forward and backward directions to create annotations for each source word. Each annotation contains summaries of both preceding and following words, focusing on words around the specific position.

The decoder generates a target word based on a context vector. Unlike the basic encoder-decoder which uses a single context vector for the whole sentence, RNNsearch uses a distinct context vector for each target word being generated. This context vector is computed as a weighted sum of the source annotations. The weight associated with each source annotation for a given target word reflects how well the input around that source word matches the output being generated at that target position. This matching score is computed by an **alignment model**, which is a feedforward neural network jointly trained with the rest of the system.

The alignment model computes a **soft alignment**, meaning the weights are probabilities indicating the importance of each source annotation when deciding the next target word. Intuitively, this implements a mechanism of **attention** in the decoder, allowing it to decide which parts of the source sentence to "pay attention to". This soft alignment allows gradients to be backpropagated through it, enabling joint training of the alignment model and the translation model. The soft alignment naturally handles source and target phrases of different lengths without needing concepts like mapping words to nowhere (NULL), unlike traditional hard alignments. Visualizations of these weights reveal how the model aligns source and target words, often showing monotonic alignment but also correctly handling non-monotonic cases like adjective-noun reordering between English and French.

Experiments were conducted on English-to-French translation using data from ACL WMT '14, comparing RNNsearch with the basic RNN Encoder–Decoder (RNNencdec). The results measured by BLEU score showed that RNNsearch significantly outperforms RNNencdec in all cases. Crucially, RNNsearch is much more robust to sentence length, showing no significant performance deterioration even with sentences of 50 words or more, while RNNencdec performance drops dramatically for longer sentences. The proposed RNNsearch approach achieved translation performance comparable or close to the conventional phrase-based system (Moses). This is notable because Moses used a larger monolingual corpus in addition to the parallel data used for training RNNsearch and RNNencdec. Qualitative analysis with examples of long sentence translations further confirms that RNNsearch translates long sentences more reliably than RNNencdec.

---

The paper concludes that the proposed RNNsearch architecture, which learns to align and translate jointly by using a soft-search mechanism over source annotations, is a promising step towards better machine translation and understanding of natural languages. This approach effectively addresses the fixed-length vector bottleneck of earlier NMT models. One remaining challenge identified is improving the handling of unknown or rare words.