Translator Apps

Language translator apps are trained on large amount of text data. They make use of Neural Machine Translation (NMT) for translating text from one language to another. A translator, whether human or computer, must frequently comprehend the entire scenario described in the source, not simply the individual words. Although all translation systems aim to model the source and destination languages, the training models used by each one of them is different.

The steps involved are:

1. **Preprocessing:** The data found in real world is hardly ideal and so preprocessing is useful to filter the data, clean the text and to prepare it for training. It involves removing the preceding and trailing spaces from a string in the dataset. Stop words i.e. the frequently used words which are without any relevant meaning or importances for training are removed. Words are shortened to their root words e.g. running is changed to run. This is helpful in finding the frequency of the words in the dataset. In texts that are taken from comments or social media, the emojis are removed, the spelling mistakes need to be corrected, and the punctuations are removed converting the whole text into lowercase. All this is achieved by using methods like:
   1. **Lower Casing:** Whole text is converted to lowercase to avoid the same words considered as different entity due to different cases.
   2. **Tokenization:** It involves breaking the paragraph into smaller units like sentences or words. Tokens can be words, phrases, or even individual characters. The most basic form of tokenization is splitting sentences into words.
   3. **Punctuation mark removal:** All the punctuations are removed from the list of words except the alphanumeric characters.
   4. **Stop Word Removal:** Stop words occur very frequently in the text but do not add any meaning to the sentence and thus are irrelevant for the training.
   5. **Stemming:** Stemming is the process of reduction of a word into its root or stem word. E.g.: running is converted to its stem word, run. This is helpful in finding the frequency of the words in the dataset.
   6. **Lemmatization:**  Stemming does not always result in words that are part of the language vocabulary. It often results in words that have no meaning to the users. In order to overcome this drawback, Lemmatization is used.
2. **Language Identification:** The first step after preprocessing is identifying the language the text is in. This is crucial in multilingual translations. Machines are trained on different models to identify the language of the source text.
3. **Translation Model Selection:** Based on the identified source and the target languages, appropriate translation model is selected. Modern translation apps often use Neural Machine Translation (NMT) models that are trained on large datasets of parallel texts in various languages.
4. **Neural Machine Models:** Neural Machine Translation is a machine translation approach that applies a large artificial neural network toward predicting the likelihood of a sequence of words, often in the form of whole sentences.
5. **Attention Mechanism:** NMT models often incorporate an attention mechanism. This helps the model to pay varying levels of attention to different parts of the source sentence while generating each word of the translation based on the relevance of the word in the source sentence.
6. **Post-processing:** The generated translation might not be completely accurate with respect to grammar and syntax. Post-processing involves making sure the grammar and syntax are correct and adapting to the conventions of the target language.
7. **Output Generation:** The translated text is generated by the app. If it is a speech-to-speech translation, the translated text may be converted to speech using text-to-speech system.
8. **Continuous Learning:** Many translation apps have the ability to continuously learn based on user feedback. This involves collecting data from user interactions and updating the existing models.

**Maintaining the structure and meaning of sentences**

Maintaining the structure and meaning of the sentence is a challenging task while translating. The translating apps use the following techniques to preserve the structure and meaning of the sentence during translation:

1. **Contextual Understanding:** Modern translation models, especially those based on neural networks are designed to capture contextual information. They analyze the entire sentence even the surrounding sentences to understand the relationships between words and phrases.
2. **Attention mechanism:** This method involves assigning attention scores to words according to their relevance in the source text. It helps to focus on different parts of the sentence at the same time while generating the output. This mechanism helps capture long-range dependencies and subtle nuances that contribute to maintaining meaning and structure.
3. **Sequence-to-Sequence Models:** These models take a sequence of words in source language and generate a corresponding sequence in the target language. By learning the alignment between source and target sequences, these models can effectively map the structure and meaning from one language to another.

Many other techniques like *subword tokenization, fine-tuning and transfer learning, post-processing rules, human feedback and iterative improvement* are also employed.

**Ways to improve the existing models**

A major challenge that remains in language translation is the sarcasm. The property of satire makes it difficult to bridge the gap between the literal and intended meaning. Sarcasm often involves saying the opposite of what is meant, relying on the context or tone and the cultural cues to convey the intended meaning. Different languages have different structures that affect how sarcasm is expressed, a literal translation may not have the same sarcastic tone. Moreover, sarcasm is deeply rooted in cultural contexts and linguistic norms and thus translating it requires deep understanding of the cultural cues.

Thus, it becomes important to train the machines to detect sarcasm and translate them into the target languages retaining the sarcastic tone of the source message. For this machines have to be trained on cross-lingual understanding and cultural sensitivity. Some of the methods that can be use to enhance the accuracy to detect sarcasm are: attention mechanisms for cross-lingual understanding, parallel sarcasm dataset creation, cross-lingual contextual understanding, language-specific sarcasm characteristics, continual learning, etc.