IJIRCCE

e-ISSN: 2320-9801 | p-ISSN: 2320-9798

# INTERNATIONAL JOURNAL OF INNOVATIVE RESEARCH

## IN COMPUTER & COMMUNICATION ENGINEERING

Volume 9, Issue 7, July 2021

INTERNATIONAL STANDARD SERIAL NUMBER INDIA

Impact Factor: 7.542

International Journal of Innovative Research in Computer and Communication Engineering

| e-ISSN: 2320-9801, p-ISSN: 2320-9798| www.ijircce.com | |Impact Factor: 7.542 |

|| Volume 9, Issue 7, July 2021 ||

| DOI: 10.15680/IJIRCCE.2021.0907206 |

# Real Time Voice Translator using Watson Language Translator

**Srinath R**

Department of Computer Science & Engineering, Velammal Engineering College, Chennai, India

**ABSTRACT:** Natural Language Processing (NLP) is a field at the intersection of computer science, artificial intelligence, and linguistics. The goal is for computers to process or "understand" natural language in order to perform tasks like Language Translation and Question Answering.With the rise of voice interfaces and chatbots, NLP is one of the most important technologies of the information age a crucial part of artificial intelligence.

Large repositories of textual data are generated from diverse sources such as text streams on the web, communications through mobile and IoT devices. Though ML and NLP have emerged as the most potent and most used technology applied to the analysis of the text and text classification remains the most popular and the most used technique. Text classification could be Multi?Level (MLC) or Multi-Class (MCC). In MCC, every instance could be assigned to only one class label, whereas MLC is a classification that assigns multiple labels to a single instance.Solving MLC problems requires an understanding of multi-label data pre?processing for big data analysis. MLC can become very complicated due to the characteristics of real-world data such as high-dimensional label space, label 11 dependency, and uncertainty, drifting, incomplete and imbalanced. Data reduction for large dimensional datasets and classifying multi-instance data is also a challenging task.The main challenge with language translation is not in translating words, but in understanding the meaning of sentences to provide an accurate translation. Each text comes with different words and requires specific language skills. Choosing the right words depending on the context and the purpose of the content, is more complicated. A language may not have an exact match for a certain action or object that exists in another language. Idiomatic expressions explain something by way of unique examples or figures of speech. Most importantly, the meaning of particular phrases cannot be predicted by the literal definitions of the words it contains.

The main challenge with language translation is not in translating words, but in understanding the meaning of sentences to provide an accurate translation. Each text comes with different words and requires specific language skills. Choosing the right words depending on the context and the purpose of the content, is more complicated. A language may not have an exact match for a certain action or object that exists in another language. Idiomatic expressions explain something by way of unique examples or figures of speech. Most importantly, the meaning of particular phrases cannot be predicted by the literal definitions of the words it contains.

## I. INTRODUCTION

To translate a corpus of English text to French or to any other languages, we need to build a recurrent neural network (RNN). RNNs are designed to take sequences of text as inputs or return sequences of text as outputs, or both. They're called recurrent because the network's hidden layers have a loop in which the output and cell state from each time step become inputs at the next time step. This recurrence serves as a form of memory. It allows contextual information to flow through the network so that relevant outputs from previous time steps can be applied to network operations at the current time step. This is analogous to how we read. As you read this post, you're storing important pieces of information from previous words and sentences and using it as context to

| e-ISSN: 2320-9801, p-ISSN: 2320-9798| www.ijircce.com | |Impact Factor: 7.542 |

|| Volume 9, Issue 7, July 2021 ||

| DOI: 10.15680/IJIRCCE.2021.0907206 |

understand each new word and sentence. Other types of neural networks can't do this (yet). Imagine you're using a convolutional neural network (CNN) to perform object detection in a movie. Currently, there's no way for information from objects detected in previous scenes to inform the model's detection of objects in the current scene. For example, if a courtroom and judge were detected in a previous scene, that information could help correctly classify the judge's gavel in the current scene, instead of misclassifying it as a hammer or mallet. But CNNs don't allow this type of time-series context to flow through the network like RNNs do.Large repositories of textual data are generated from diverse sources such as text streams on the web, communications through mobile and IoT devices. Though ML and NLP have emerged as the most potent and most used technology applied to the analysis of the text and text classification remains the most popular and the most used technique. Text classification could be Multi?Level (MLC) or Multi-Class (MCC). In MCC, every instance could be assigned to only one class label, whereas MLC is a classification that assigns multiple labels to a single instance. Solving MLC problems requires an understanding of multi-label data pre?processing for big data analysis.

## II.MACHINE TRANSLATION

Machine Translation (MT) is a subfield of computational linguistics that is focused on translating text from one language to another. With the power of deep learning, Neural Machine

Translation (NMT) has arisen as the most powerful algorithm to perform this task. While Google Translate is the leading industry example of NMT, tech companies all over the globe are going all in on NMT.This state-of-the-art algorithm is an application of deep learning in which massive datasets of translated sentences are used to train a model capable of translating between any two languages. With the vast amount of research in recent years, there are several variations of NMT currently being investigated and deployed in the industry. One of the older and more established versions of NMT is the Encoder Decoder structure. This architecture is composed of two recurrent neural networks (RNNs) used together in tandem to create a translation model. And when coupled with the power of attention mechanisms, this architecture can achieve impressive results.

## III.TRANSLATION MODEL

To translate a corpus of English text to French or to any other languages, we need to build a recurrent neural network (RNN). RNNs are designed to take sequences of text as inputs or return sequences of text as outputs, or both. They're called recurrent because the network's hidden layers have a loop in which the output and cell state from each time step become inputs at the next time step. This recurrence serves as a form of memory. It allows contextual information to flow through the network so that relevant outputs from previous time steps can be applied to network operations at the current time step.

This is analogous to how we read. As you read this post, you're storing important pieces of information from previous words and sentences and using it as context to understand each new word and sentence.

Other types of neural networks can't do this (yet). Imagine you're using a convolutional neural network (CNN) to perform object detection in a movie. Currently, there's no way for information from objects detected in previous scenes to inform the model's detection of objects in the current scene. For example, if a courtroom and judge were detected in a previous scene, that information could help correctly classify the judge's gavel in the current scene, instead of misclassifying it as a hammer or mallet. But CNNs don't allow this type of time-series context to flow through the network like RNNs do.

International Journal of Innovative Research in Computer and Communication Engineering

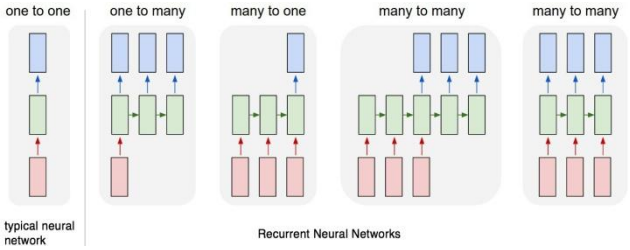
| e-ISSN: 2320-9801, p-ISSN: 2320-9798| www.ijircce.com | |Impact Factor: 7.542 |

|| Volume 9, Issue 7, July 2021 ||

| DOI: 10.15680/IJIRCCE.2021.0907206 |

## IV.RNN SETUP

Depending on the use-case, you'll want to set up your RNN to handle inputs and outputs differently. For this project, we'll use a many-to-many process where the input is a sequence of English words and the output is a sequence of French words (fourth from the left in the diagram below).




Each rectangle is a vector and arrows represent functions (e.g. matrix multiply). Input vectors are in red, output vectors are in blue and green vectors hold the RNN's state (more on this soon).

From left to right: (1) Vanilla mode of processing without RNN, from fixed-sized input to fixed-sized output (e.g. image classification). (2) Sequence output (e.g. image captioning takes an image and outputs a sentence of words). (3) Sequence input (e.g. sentiment analysis where a given sentence is classified as expressing positive or negative sentiment). (4) Sequence input and sequence output (e.g. Machine Translation: an RNN reads a sentence in English and then outputs a sentence in French).

(5) Synced sequence input and output (e.g. video classification where we wish to label each frame of the video). Notice that in every case there are no pre-specified constraints on the lengths sequences because the recurrent transformation (green) is fixed and can be applied as many times as we like.

## V.PRE-PROCESSING

Here is a sample of the data. The inputs are sentences in English; the outputs are the corresponding translations in French.

```
English sample 1:  new jersey is sometimes quiet during autumn , and it is snowy in april .
French sample 1:  new jersey est parfois calme pendant l' automne , et il est neigeux en avril .

English sample 2:  the united states is usually chilly during july , and it is usually freezing in november .
French sample 2:  les états-unis est généralement froid en juillet , et il gèle habituellement en novembre .

English sample 3:  california is usually quiet during march , and it is usually hot in june .
French sample 3:  california est généralement calme en mars , et il est généralement chaud en juin .

English sample 4:  the united states is sometimes mild during june , and it is cold in september .
French sample 4:  les états-unis est parfois légère en juin , et il fait froid en septembre .

English sample 5:  your least liked fruit is the grape , but my least liked is the apple .
French sample 5:  votre moins aimé fruit est le raisin , mais mon moins aimé est la pomme .
```

When we run a word count, we can see that the vocabulary for the dataset is quite small. This was by

| e-ISSN: 2320-9801, p-ISSN: 2320-9798| www.ijircce.com | |Impact Factor: 7.542 |

|| Volume 9, Issue 7, July 2021 ||

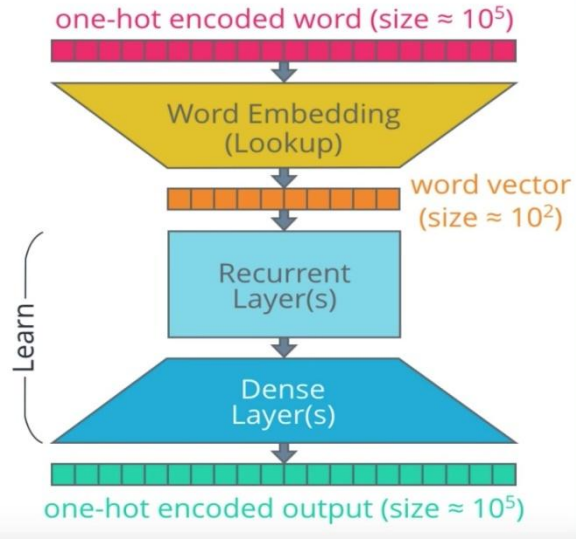
| DOI: 10.15680/IJIRCCE.2021.0907206 |

design for this project. This allows us to train the models in a reasonable time.Next, we need to tokenize the data — i.e., convert the text to numerical values. This allows the neural network to perform operations on the input data. For this project, each word and punctuation mark will be given a unique ID. (For other NLP projects, it might make sense to assign each character a unique ID.).When we run the tokenizer, it creates a word index, which is then used to convert each sentence to a vector.

When we feed our sequences of word IDs into the model, each sequence needs to be the same length. To achieve this, padding is added to any sequence that is shorter than the max length (i.e. shorter than the longest sentence).One-Hot Encoding (not used)

```
Sequence 2 in x
  Input:  [10 11 12  2 13 14 15 16  3 17]
  Output: [10 11 12  2 13 14 15 16  3 17] no padding
Sequence 3 in x
  Input:  [18 19  3 20 21]
  Output: [18 19  3 20 21  0  0  0  0  0]  padding
```

In this project, our input sequences will be a vector containing a series of integers. Each integer represents an English word (as seen above). However, in other projects, sometimes an additional step is performed to convert each integer into a one-hot encoded vector. We don't use one-hot encoding (OHE) in this project, but you'll see references to it in certain diagrams (like the one below).



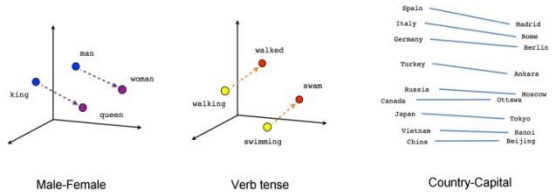

Embeddings allow us to capture more precise syntactic and semantic word relationships. This is achieved by projecting each word into n-dimensional space. Words with similar meanings occupy similar regions of this space; the closer two words are, the more similar they are. And often the vectors between words represent

| e-ISSN: 2320-9801, p-ISSN: 2320-9798| www.ijircce.com | |Impact Factor: 7.542 |

|| Volume 9, Issue 7, July 2021 ||

| DOI: 10.15680/IJIRCCE.2021.0907206 |

useful relationships, such as gender, verb tense, or even geopolitical relationships.

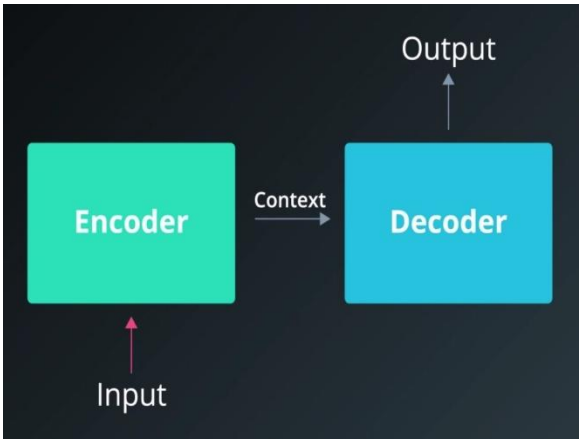



Male-Female Verb tense Country-Capital

Training embeddings on a large dataset from scratch requires a huge amount of data and computation. So, instead of doing it ourselves, we'd normally use a pre-trained embeddings package such as GloVe or word2vec. When used this way, embeddings are a form of transfer learning. However, since our dataset for this project has a small vocabulary and little syntactic variation, we'll use Keras to train the embeddings ourselves.

Encoder & Decoder

Our sequence-to-sequence model links two recurrent networks: an encoder and decoder. The encoder summarizes the input into a context variable, also called the state. This context is then decoded and the output sequence is generated.




Since both the encoder and decoder are recurrent, they have loops which process each part of the sequence at different time steps. To picture this, it's best to unroll the network so we can see what's happening at each time step.

In the example below, it takes four timesteps to encode the entire input sequence. At each time step, the encoder "reads" the input word and performs a transformation on its hidden state. Then it passes that hidden state to the next time step. Keep in mind that the hidden state represents the relevant context flowing through the network. The bigger the hidden state, the greater the learning capacity of the model, but also the greater the computation requirements. We'll talk more about the transformations within the hidden state when we cover gated recurrent units (GRU).

For now, notice that for each time step after the first word in the sequence there are two inputs: the hidden

| e-ISSN: 2320-9801, p-ISSN: 2320-9798| www.ijircce.com | |Impact Factor: 7.542 |

|| Volume 9, Issue 7, July 2021 ||

| DOI: 10.15680/IJIRCCE.2021.0907206 |

state and a word from the sequence. For the encoder, it's the next word in the input sequence. For the decoder, it's the previous word from the output sequence.

Also, remember that when we refer to a "word," we really mean the vector representation of the word which comes from the embedding layer.Here's another way to visualize the encoder and decoder, except with a Mandarin input sequence.

Bidirectional Layer

Now that we understand how context flows through the network via the hidden state, let's take it a step further by allowing that context to flow in both directions. This is what a bidirectional layer does.

In the example above, the encoder only has historical context. But, providing future context can result in better model performance. This may seem counterintuitive to the way humans process language since we only read in one direction.
However, humans often require future context to interpret what is being said. In other words, sometimes we don't understand a sentence until an important word or phrase is provided at the end.To implement this, we train two RNN layers simultaneously.

Hidden Layer with Gated Recurrent Unit (GRU).Now let's make our RNN a little bit smarter. Instead of allowing all of the information from the hidden state to flow through the network, what if we could be more selective? Perhaps some of the information is more relevant, while other information should be discarded. This is essentially what a gated recurrent unit (GRU) does.

## VI.CONCLUSION

One of the long-standing promises of modern technology has been the ability to translate conversations in real-time. Google is making this a reality with a feature called Interpreter mode. Previously available on Google Home and Nest devices, it will now work on any Android or iOS phone running the Google Assistant. A more useful place for it, if you ask me. Simply say "Hey Google, help me speak Spanish" or "be my Spanish translator," and Google will help you out with speedy translations that are both displayed on-screen and read out loud. Google will also provide a few Smart Replies below the translation to help you move along the conversation more quickly. You can also simply use the keyboard if you're in a quiet environment. Real-time voice translation is equally incredible, and can act as an intermediary for two people holding a conversation using different languages. You tap the in-app mic once and starting talking in the foreign tongue first. Then once the first language has been recognized — tap the mic again and both people can begin talking. The app pulls in text-based translations of both sides of the conversation in real-time, helping overcome the language barrier. This will work live for speeches, lectures, and other spoken word events and from pre?recorded audio, too. That means you could theoretically hold your phone up to computer speakers and play a recording in one language and have it translated into text in another without you having to input the words manually.

## VII.FUTURE WORK

In future, the application will be improved. New features like two factor authentications will be added. It will be converted into a real time chat application. The number of languages could be used will be increased.

| e-ISSN: 2320-9801, p-ISSN: 2320-9798| www.ijircce.com | |Impact Factor: 7.542 |

|| Volume 9, Issue 7, July 2021 ||

| DOI: 10.15680/IJIRCCE.2021.0907206 |

## REFERENCES

[1]Unsupervised Neural Machine Translation With Cross-Lingual Language Representation AgreementProfessor Haipeng Sun – 2018
[2]On Application of Natural Language Processing in Machine Translation - Zhaorong Zong;Changchun Hong 2018 3rd International Conference on Mechanical, Control and Computer Engineering (ICMCCE)
[3]Dual Translation of International and Indian Regional Language using Recent Machine Translation - 2020 3rd International Conference on Intelligent Sustainable Systems (ICISS)
[4]A new Chinese-English machine translation method based on rule for claims sentence of Chinese patent - 2011 7th International Conference on Natural Language Processing and Knowledge Engineering
[5] Automatic evaluation methods of a speech translation system's capability - IEEE Workshop on Automatic Speech Recognition and Understanding, 2001. ASRU '01.
[6] A survey of voice translation methodologies — Acoustic dialect decoder - 2016 International Conference on Information Communication and Embedded Systems (ICICES)
[7] Multilingual speech to speech translation system in bluetooth environment - 2014 International Conference on Control, Instrumentation, Communication and Computational Technologies (ICCICCT).
[8] Speech Recognition for Voice-Based Machine Translation - IEEE Software ( Volume: 31, Issue: 1, Jan.-Feb. 2014) 58
[9] On Application of Natural Language Processing in Machine Translation - 2018 3rd International Conference on Mechanical, Control and Computer Engineering (ICMCCE)
[10] Automatic evaluation methods of a speech translation system's capability - IEEE Workshop on Automatic Speech Recognition and Understanding, 2001. ASRU '01.
[11] Statistical Machine Translation System for Indian Languages - 2016 IEEE 6th International Conference on Advanced Computing (IACC)
[12] Semantic natural language translation based on ontologies combination - 2017 8th International Conference on Information Technology (ICIT)
[13] A survey of voice translation methodologies — Acoustic dialect decoder Publisher: IEEE - 2020 International Conference on Information Communication and Embedded Systems (ICICES), Infrastructure (PKI). Karabey, Gamze Akman. 16 February 2017.
[14] https://github.com/IBM/watson-speech-translator/find/master
115] https://speech-to-text-demo.ng.bluemix.net

INNO SPACE
SJIF Scientific Journal Impact Factor
Impact Factor: 7.542

doi cross ref

ISSN INTERNATIONAL STANDARD SERIAL NUMBER INDIA

निस्केयर NISCAIR

# INTERNATIONAL JOURNAL OF INNOVATIVE RESEARCH

## IN COMPUTER & COMMUNICATION ENGINEERING

9940 572 462 6381 907 438 ijircce@gmail.com



Scan to save the contact details