International Research Journal of Engineering and Technology (IRJET) e-ISSN: 2395-0056
Volume: 06 Issue: 11 | Nov 2019 www.irjet.net p-ISSN: 2395-0072

On-screen Translator using NLP and Text Detection

Prof. Vidya G. Shitole[1], Mr. Aditya Agashe[2]

[1]Assistant Professor, Suman Ramesh Tulsiani Technical Campus, FoE – Kamshet, Maharashtra, India.
[2]Student, Suman Ramesh Tulsiani Technical Campus, FoE – Kamshet, Maharashtra, India

---

Abstract - *Newer is the language, hard to understand new words. If you get stuck at any point while reading a paragraph to translate something in the understandable language you need a translation of that particular word. Instead of going on to the new tab and searching for the translation of the word, if the translation is available on-screen over that word will be comfortable for a person. To translate a word or phrase into a native language from a different language we can use the application of Natural Language Processing and it can be made reliable using text detection technique. NLP has a syntax checker to check the grammar of input and also to rebuild input. NLP can be used to format a scattered text. Text detections fetch the text and pass parameters of it to methods to search for the translated form of the input text. It can be from a video frame or webpage. Text can be fetched using OCR (optical character recognition) method from sensors or by using input images or sources, that to be translated.*

***Key Words*: NLP, text detection, Natural language processing, OCR,**

1. INTRODUCTION

Machine translation depends on a large dataset of sentences or words of both source and target language. Monolingual data is widely used to train language models which increases the performance and fluency of machine translation. We are mainly focusing on back-translation because it operates in a semi-supervised setup. Here both monolingual and bilingual data in target language are available. Detection of text, fetching content from detection, processing input and proving appropriate result as an output will be the format of the procedure.

Text detection will take place from the source input like images or frame or text selection, after that we need to read the source text to give it as input to the processing model. After that to process input, we have a model that will use LSTM (Long Short-Term Memory) and RNN to analyse the word from a dataset and which will return the predicted suggestion as an output which will be our final result of the program.

1.1 Text Detection

Text detection is basically a recognition of text from the image using various tools. We are passing the images as input from the frame as an image format. Here we are using a Python-tesseract tool.

1.2 NLP (**N**atural **L**anguage **P**rocessing)

Natural language processing is a very large area in machine learning and deep learning field. It works in the format of the input sequence – MODEL – output sequence.

1 Back translation

2 Byte-Pair encoding or Character-based

3 Automatic Post Editing

4 Parallel filtering

5 Deep Word Representation

2. Literature Review

Extracting information is concerned to identify similar relevant phrases in textual data. For many applications, extracting entities such as names, places, events, dates, times and prices is a powerful way of summarizing the information relevant to a user's needs. In the case of a domain-specific translation model, the automatic identification of important phrases can increase the accuracy and efficiency of a translation. Phrase detection and backtracking in NLP translation will help in organizing the whole process. OCR text detection will occur as the user give input to the system manually but the translation to the base language will take place automatically.

The text detection process is a stepwise process. First, we take an appropriate image input so that it will fit dimensionally in a model for further processes. Then to perform text detection, we use a deep learning model to extract to find out output feature maps from two layers. The first layer is output sigmoid activation which will tell us the probability of region containing text or not. The second output feature map layer will help to decide to bound of text area in an image. This will provide us a frame or window around a text area bounds so that it will mark up the area and traces the text. Now next we take that frame as input and our model will try to grayscale the text area to check traces of higher scores. It will try to match a pattern with a dataset and it will tokenize the word and check in a dataset for appropriate match and return the tokens and using those we will get the result as the translated text.

3. System information and architecture

This system is mainly designed to help people in translating non-familiar words from other languages to the target

language (English). The use of RNN and LSTM will help in increasing machine translation source language to target language.

Procedure starts from the detection of text, we use OpenCV for that; approach of OpenCV generally claims:

1) apply filters to highlight characters

2) to recognize characters one by one contour detection is applied

3) to identify characters, apply image classification

Mostly we work on unstructured data, so there are some prerequisites before moving towards the model-building part. We take input from an image-text to string text. Currently, we are using Python-tesseract a tool for OCR, which is used to extract embedded text from images. Then we will convert that object into a string and pass that to the model to normalize and tokenize.

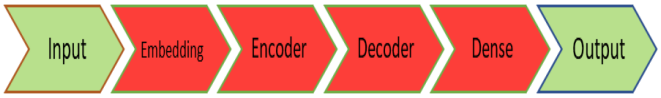

Fig: LSTM layer followed by dense layer

Then the model will treat that string according to the rules defined in the model description and layers to format the fetched string. The main requirement of the Seq2Seq model is, it requires both input and output sentences to be converted into an integer sequence of fixed length. But before we do that, let's visualize the length of the sentences. We will capture the lengths of all the sentences in two separate lists for target language and source language, respectively. Then the model will use LSTM to check the correct relation between previous and next word relationships to return a correct token and then the model will return the translation suggestion and that will be our output.

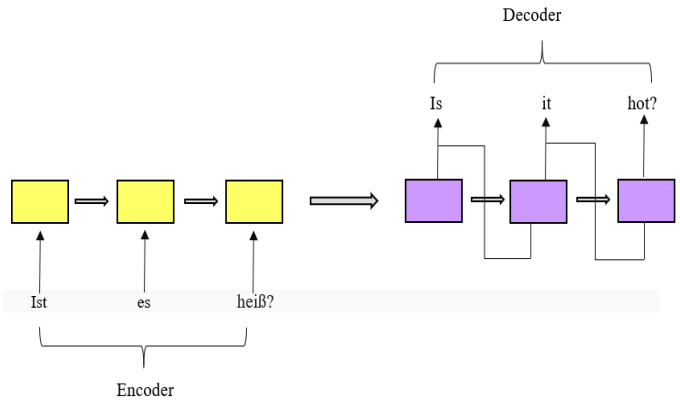

Fig: Two different RNN

Here, both, the input and output are sentences. In other words, these sentences are the input and output parameters of a model. This is the basic concept of Sequence-to-Sequence modeling.

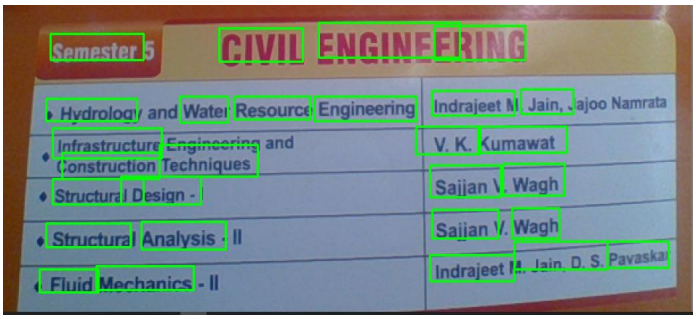

Fig: text detection in a video frame

Here in the above figure, we are using the EAST model to detect text from a video frame. Video captures text at 13 FPS and 720p resolution.

The EAST (An Efficient and Effective Accurate Scene Text Detector) achieves approximately 78% F-score. The model is a fully convolutional neural network made for text detection that will give dense per-pixel predictions of words or text as an output. The post-processing includes the network management system and thresholding on predicted geometric shapes.

NLP individual models:

1) *p(f|e)* Translation model

-due to Bayes algorithm reverse ordering of f and e

-English sentences having the same meaning as the foreign sentence will be assigned to the higher probability

-a bilingual (parallel) corpus for estimation is needed

2) *p(e)* The language model

-to fluent/grammatical sentences higher probability is assigned

-monolingual corpus is needed for estimation

Translation model

Enter alignment variable '*a*' which represents alignments between the individual words in the sentence pair

$$p(f|e) = P\ \boldsymbol{a}\ p\ (\boldsymbol{a}, f|e)$$

Alignment probabilities

Breaking of sentence into a manageable chunks or words

$$p(a, f|e) = \prod_{j=1}^{m} t(f_j|e_i)$$

where '*ei'* is the English word(s) corresponding to the French word *'fj'* and '*t(fj|ei)'* is the (conditional) probability of the words being aligned.

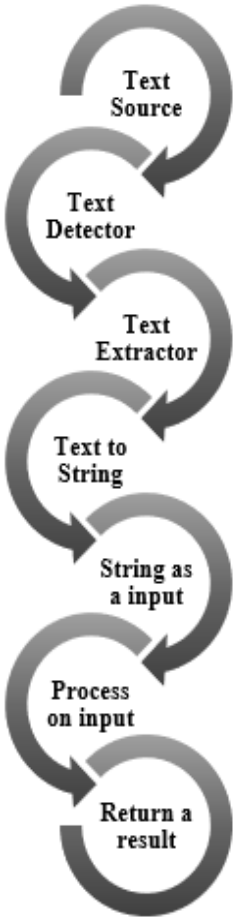

Fig: System work-flow diagram

The system starts working from taking input as an image, it may be from an image or video frame.

Then using a tool to extract corpus of text frame from image then converting text to string is the whole procedure to make model acceptable ready input. Afterward, the NLP model will analyze the string, tokenize and then it will return the resulting translation.

4. Future Scope

Translator is a thing that is useful to everyone who reads and writes. Deploying as an extension to the web browser and making available to all will help a lot to the new language learners. There are many scopes of improvements to this system, and definitely making this system faster will be our main focus. How to make this is a thing available offline to use is also a part of future development and adding add-on support of multiple regional languages for translation will be the update part of it.

Possible directions for future research include: (1) Changing geometry formulation to detect directly curved / curvature text (2) Text detector + text recognizer integration (3) general object detection and suggestions.

4. CONCLUSIONS

Detecting text using various tools and methods, analyzing source language and then processing that to return a translated word; Hence, we have designed an on-screen translator system using NLP and there are scopes for improvements. Better use of the application of neural networks can be seen in this application.

Fully convolutional networks are fast, end-to-end models for pixel-wise problems and it helped us a lot.

REFERENCES

[1] EAST: An Efficient and Accurate Scene Text Detector "Xinyu Zhou, Cong Yao, He Wen, Yuzhi Wang, Shuchang Zhou, Weiran He, Jiajun Liang", Submitted on 11 Apr 2017

[2] Foundations of Statistical Natural Language Processing, Manning and Schutze, ch. 13

[3] Mobile Camera Based Text Detection and Translation," Derek Ma, Qiuhau Lin, Tong Zhang".

[4] http://code.google.com/p/tesseract-ocr/

[5] https://www.pyimagesearch.com/

[6] Natural Language Processing and Machine Translation Encyclopedia of Language and Linguistics, 2nd ed. (ELL2). Machine Translation: Interlingual Methods, "Bonnie J. Dorr, Eduard H. Hovy, Lori S. Levin"

[7] Understanding Back-Translation at Scale, "Sergey Edunov, Myle Ott, Michael Auli, David Grangier".

[8] Google ML and deeplearning.ai