

An Effective Hybrid Approach Based on Machine Learning Techniques for Auto-Translation: Japanese to English

Mhd Saeed Sharif
School of Architecture, Computing and
Engineering, ACE, UEL, University Way,
London, UK.

Bilyaminu Auwal Romo
School of Architecture, Computing and
Engineering, ACE, UEL, University Way,
London, UK.

Harry Maltby
School of Architecture, Computing and
Engineering, ACE, UEL, University Way,
London, UK.
u1606661@uel.ac.uk

Ali Al-Bayatti
School of Computer Science and Informatics,
CEM, De Montfort University, Leicester, LE1
9BH, UK

Abstract—In recent years machine learning techniques have been able to perform tasks previously thought impossible or impractical such as image classification and natural language translation, as such this allows for the automation of tasks previously thought only possible by humans. This research work aims to test a naïve post processing grammar correction method using a Long Short Term Memory neural network to rearrange translated sentences from Subject Object Verb to Subject Verb Object. Here machine learning based techniques are used to successfully translate works in an automated fashion rather than manually and post processing translations to increase sentiment and grammar accuracy. The implementation of the proposed methodology uses a bounding box object detection model, optical character recognition model and a natural language processing model to fully translate manga without human intervention. The grammar correction experimentation tries to fix a common problem when machines translate between two natural languages that use different ordering, in this case from Japanese Subject Object Verb to English Subject Verb Object. For this experimentation 2 sequence to sequence Long Short Term Memory neural networks were developed, a character level and a word level model using word embedding to reorder English sentences from Subject Object Verb to Subject Verb Object. The results showed that the methodology works in practice and can automate the translation process successfully.

Keywords—*Optical Character Recognition (OCR), Artificial Intelligence (AI), Natural Language Tool Kit (NLTK), Long Short Term Memory (LSTM), Bilingual Evaluation Understudy (BLEU).*

I. INTRODUCTION

The problem at hand is that currently machine translation often suffers from a common problem where the natural language ordering of one language is kept after being translated to another. This is most evident in languages that are very distant from one another such as between English and Japanese. A system that can translate manga (Japanese comics) from the Japanese language into English is used to give the experiment context and to experiment with bounding box object detection to identify text regions within each page.

Machine learning techniques are fully capable of performing all the steps to do so if modeled in a systematic manner, using bounding box object detection [1], OCR [2] and machine translation [3]. The novel problem of translating manga from Japanese to English can be solved once broken down into 3 smaller tasks, identifying the image coordinates of the text to be translated using bounding box object detection, reading the text in a form that a computer can understand and then translating the text. From here the process can just be run in reverse converting the text into an image and then placing the newly translated text over origin of said text.

Translating works between languages can be very mentally intensive for bilingual individuals, and even for people using tools such as Google Translate can take a considerable amount of time. Due to this many works do not get translated to the point where fans take it upon themselves to translate works [4]. Using machines to perform the translations would help works get translated as it would remove many of the limiting factors that prevent works from being translated. These were all motivations for the case study.

A system was created using the proposed methodology that can identify text within a page and locate the coordinates. Using these coordinates a smaller image was produced as to extract the text reliably. This smaller image then had the text extracted from it in the form of UTF8. This text was then be translated from Japanese to English. The text was then placed into an image and placed back where it was in the original image using the coordinates gained from the object detection in the first stage. This process was repeated for each textbox on the page.

This 3-part methodology has been used before to moderate success as seen in a GitHub repository by user Dalelyunas [5]. In that instance the text boxes within the target page were identified using OpenCV contour manipulations. Their implementation of text detection first converts the target image to black and white using a threshold, secondly the image is eroded to remove the small contours within the image, thirdly a

mask is created with the text contours coordinates ready for the translation stage.

Attempts at a naive post processing technique were used to try and overcome a common problem when translating between languages that do not share the same natural order. A neural network was trained to reorder and add or remove words after they have been translated by a translation model. The English language uses the natural ordering of SVO whereas the Japanese language uses the natural ordering SOV. This presents a problem for neural machine translation systems as the system needs to learn another dimension to the text while also learning the translations for words. The end goal of doing this is to see whether this automated translation methodology will work in practice and analyse the BLEU score of a model trained to reorder English grammar to see if LSTM networks [6] can reorder SOV to SVO [7] and show whether this technique is statistically worthwhile and effective [8].

II. BACKGROUND RESEARCH & LITERATURE REVIEW

A. AI and Machine Learning

AI is the very broad field that grants machines the ability to make decisions, and in turn effective intelligence in given fields. Intelligence was described in [9] as the ability to think and understand instead of doing things by instinct or automatically. In contrast the definition of AI given in IEEE journal of oceanic engineering [10] provides a more broad definition ‘The term artificial intelligence denotes behaviour of a machine which, if a human behaves in the same way, is considered intelligent.’ AI is often classified into 2 separate categories, specific AI that has intelligence in a specific intelligence domain, for example a chess AI is only knowledgeable in the domain of chess. The other would be a general artificial intelligence, this would have a general or at least basic understanding of different intelligence domains once trained. The latter would be more similar to a human being, being able to reply with coherent answers and actions to given questions and tasks. In 1950 Alan Turing proposed a test, now known as the ‘Turing test’ that tests to see whether a human interrogator can tell the difference between 2 subjects, a human and a machine [11]. Both subjects respond to the interrogator by unbiased means so that the interrogator cannot tell who is who by looking at the communication mediums, handwriting for example. The Turing test is often used in modern times when testing chatbots or with automated personal assistants such as Google’s Duplex [12].

Machine learning (ML) involves teaching algorithms how to make decisions and solve problems by training them using the data from previous outcomes or targets, and thus giving them forms of intelligence. Using modern techniques provides machines with intelligence in specific domains of which the data has been derived. Machine learning is different from traditional programming as it involves programming a machine to ‘train’ itself to solve a problem and feeding the program data about previous decisions and outcomes. Conventional programming seeks to build a machine to solve a problem and produce decisions and outcomes. Training ML based algorithm has two main difficulties, the first one is that training a machine to make reliable and robust decisions requires a lot of training epochs and testing data. Acquiring this training and testing data can take a lot of human work hours as each piece of data needs to be

labelled with the exception of unsupervised learning. This is mainly a problem when it is very hard or impossible to gain more data, an example of this would be an image classification system to classify very rare animals. Without enough training data the model will most likely overfit and not be capable of providing robust outputs. The second problem is the time and or processing power required to process the system while it is training. Large deep neural networks can take days to train on a powerful hardware currently available. An example would be some of the deep convolutional neural network algorithms used in the ImageNet competition [13, 14].

B. Importance of Data

The clarity and representation of the data can make large differences in the outcome of the system making it very chaotic in nature. This statement can be proven by observing the outcomes of machine learning experiments where the model is the control variable and the data used to train the model is the independent variable. By using different representations of given real world data, a model would find different patterns and thus produce different results. The model will always try to produce outputs relative to what it’s being shown, showing the model the wrong data will lead to it producing inaccurate and undesired outputs. Negnevitsky in [9] states 3 main reasons for errors in data:

- I. Incompatible data, normally arising from data that has been processed into different forms.
- II. Missing data entries, could be intended or unintended. Negnevitsky’s example is that not having a business phone may imply they are unemployed.
- III. Inconsistent data, inconsistent forms for the data. Usually the worst since it can make the data unusable until it is cleaned.

C. Convolutional Neural Networks

Convolutional neural networks extract features from a set of data being put into the network and place them into a feature map. They do this with the use of pooling layers, which use filters to extract patterns. The network effectively moves a small grid across an image and uses a predefined filter logic such as only taking the highest value as shown in Figure 1 or taking the average value. This process can be performed over and over to obtain more abstract features. A convolutional neural network usually ends with a standard neural network layer to extract the patterns within the feature maps produced from the input data.

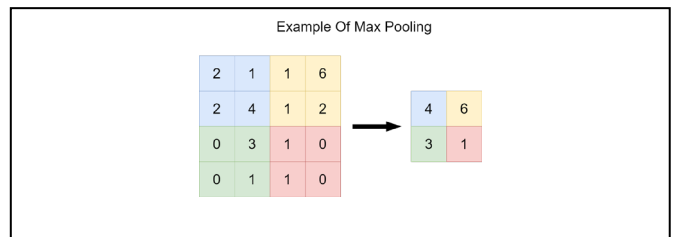


Fig. 1. Graphic to illustrate max pooling feature extraction within a convolutional neural network.

D. Recurrent Neural Networks

Recurrent neural networks feed the output of the network back into the network. LSTM has proven to be very effective in

the domain on natural language processing and has managed to overcome the vanishing gradient/exploding gradient that normal recurrent neural network architectures are prone to. Another benefit to the LSTM is that it can ‘forget’ certain inputs and effectively choose to only remember important parts of the input data. Due to this all modern real-world applications of recurrent neural networks use some variation of the LSTM, an example of one that is used today is the GRU which is a less computationally intensive application of the LSTM concept [15].

E. Object Detection Models

In the research work [16] object detection models utilizing Google’s object detection API have shown very good results in detecting objects within an image. Recent models that have had their features detectors trained on the iNaturalist Detection Dataset have provided great results in detecting objects with limited data. While [17] outlines the effectiveness of a recurrent neural network autoencoder to learn the encoded meaning of phrases in the context of neural machine translation. They show that their model can learn both the semantics and syntax of the linguistic phrases fed to the network. They also go on to describe their results using hidden LSTM units while they emphasize the importance of the forget gate within the LSTM.

To evaluate machine translation systems a method was proposed mid 2002 by [14]. This method is called Bleu and can evaluate machine translations against a professional human translation. This method is still used today due to the simplicity and robustness of this method. Many people do not use the method directly but instead create a deviation of the algorithm for their specific use case. As this method has become a de facto standard it will be used to gauge the performance of the experiments if the experiments produce viable results. In other research work, modern translation systems utilise auto encoders to encode words and phrases using an encoder that has been trained on the input language to produce vectors. These vectors are then decoded using a decoder trained on the target language [19, 20]. This uses the principle that words that hold similar sentiment should be encoded into similar vectors. These vectors are effectively a universal language and as such any word that holds similar meanings should be translated to similar vectors. With this concept a naive solution to grammar correction in translated text will be tested.

F. Encoding words into vectors using Word2Vec

Word2Vec is a tool that has been developed using the principle of turning words into vectors [19]. The tool uses techniques described in [19] to be able to create these vectors. The tool reads a corpus and based on factors such as the positioning and frequency of the words affect how the words are turned into vectors. Very interestingly algebraic operations can be performed on these vectors showing that word vector similarity goes beyond syntactic regularities. An example in the paper [18], if you take away the vector for ‘man’ from the vector for ‘king’ and then add the vector for ‘woman’, the result is a very similar vector encoding for the word ‘queen’. Having these types of encoding instead of 1 hot encoding allow for the analysis of how words relate to each other and enable the vector representations to be manipulated in respect to each other with arithmetic operations.

G. Constituent reordering and syntax models from English to Japanese

The research paper [20] describes a pre-processing technique that parses the constituent parts of the input sentence and provides an increase in their BLEU score of 4.76 using at the time the most advanced techniques. The source sentence string is converted into a tree data structure and allows for the source sentence to be reordered, one of their main focuses was to reorder the source English sentence which is considered largely head initial into an ordering that fits in line with the Japanese head final ordering. This is very similar to what was attempted in our research.

III. ARCHITECTURE OF THE SOLUTION AND IMPLEMENTING THE METHODOLOGY

A. Architecture of the Solution

Python was used as the common programming language for the research, as it is very versatile and has been used frequently in the area of machine learning. Due to this, many frameworks have been made to interface with Python. All the components of the system were scripted together using Python.

The methodology uses three parts, the first part is a bounding box object detection model to detect Japanese text on the page and return the coordinates. The second part is an OCR model to convert the text from image form to Unicode character form, the coordinates from the bounding box model are used to create a cropped image. The third step is the machine translation part where the Japanese Unicode text is translated into English Unicode text. The process is then reversed by converting the English Unicode into image form and the placing it back where the coordinates said replacing the Japanese text and translating the scanned manga page. This methodology is illustrated in Figure 2.

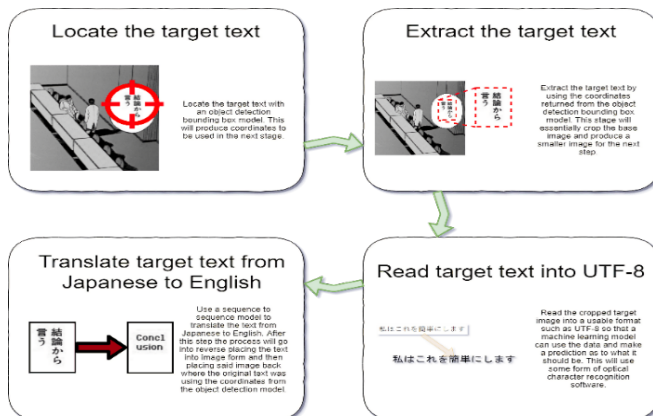


Fig. 2. Graphic to illustrate the proposed method for translating manga in an automated fashion.

B. Object Detection

This part of the system needs to locate the text to be translated within the manga image and provide the coordinates of the text so that it can later be cropped, reducing the noise of the target image and making it possible to translate as the target text has been isolated.

From the literature review it had been noted that TensorFlow bounding box models produce impressive results in terms of

image categorization and false positives (The iNaturalist Species Classification and Detection Dataset), [16].

A naive approach was taken where an object recognition model was trained to identify areas with text within pages of manga and draw the bounding boxes around text so the coordinates from the boxes can be used.

For the bounding box part of the implementation, an open source TensorFlow model was used. It was trained on 360 images which had been hand labelled and split into 75% training and 25% testing. The 360 images were from the first and second chapter of the chosen manga *Give My Regards to Black Jack* by Shūhō Satō available at Manga On Web [21]. The model used is called *faster_rcnn_inception_v2_coco* and is available at the TensorFlow model zoo [22] on GitHub. The model is permissible for use under the Apache license allowing for commercial use, modification, distribution, patent use and private use. The model’s feature extraction has been trained on the *coco dataset* which is a dataset of various mundane objects making it good for general purpose object detection. Labelling the dataset was quite a time-consuming task as labelling and saving each image would take 1 to 2 minutes.

To create this dataset, we integrated LabelImg [23] into our approach. This program allowed for the creation of xml files for the object detection model after loading an image into the program. This allows the user to label the text by drawing bounding boxes around the object desired to be detected, in this case Japanese text.

The manga used for this task is a manga named *Give My Regards to Black Jack* which the creator had released into the public domain after he reached 10 million sales [24 - 26] allowing it to be used. TensorFlow for the API was also used [27]. Figure 3 shows the training graph for *faster_rcnn_inception_v2_coco* when training to detect Japanese text boxes.

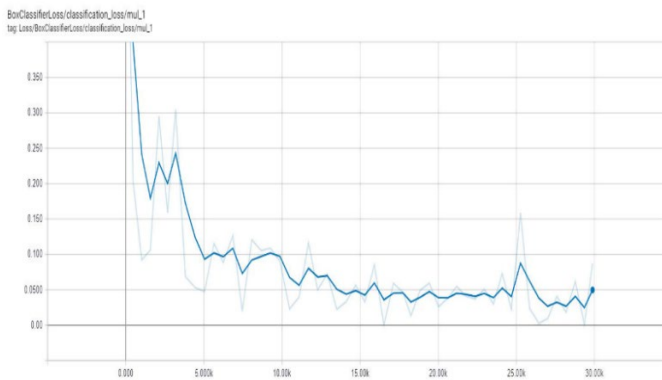


Fig. 3. Graph showing the loss over time as the object detection model was training.

The boxes drawn in figure 4 are being drawn from coordinates calculated by the object detection model, the boxes are for debugging and illustrative purposes as well as providing a qualitative test result. Some of the boxes report a 100% figure, in most other instances this may be a sign that the model has overfit or there is a problem where the model is generalising too little. Which could also mean that these images are not a valid report of the accuracy. But in this case where the model has been

trained to read text this is probably not the case as the individual characters will always look the same as they use the same font meaning the model has actually seen the same character before giving a 100% accuracy report. It is acknowledged that this would not be the case if the model were to be provided with manga using a different font but if the model has been trained well on a variety of fonts it should

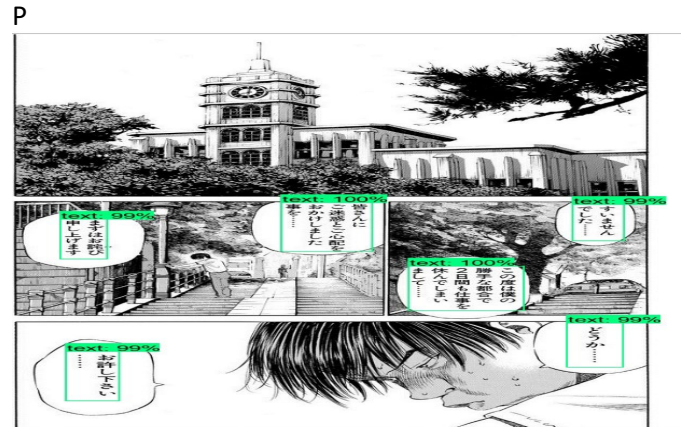


Fig. 4. A test image to provide a qualitative example of object detection reliability when detecting text regions [21].

C. Optical Character Recognition

For the OCR part of the system *Tesseract OCR* was employed and integrated. The principle technologies used in the software were conceived by Dell. This method for OCR is open source and easy to use as well as being very versatile with high performance due to its novel line finding and feature/classification techniques [2].

Tesseract OCR was chosen as a solution as it has a very good history and is currently being maintained by Google, From the literature review which contains an overview paper on Tesseract OCR, Tesseract became the first choice. This solution can easily implement Python wrapper that would integrate with the code. Afterwards, Tesseract API is called through the wrapper and importing the *tesseract* (Tesseract OCR) module and passing it the cropped image and a language weight file to use. The weight file for reading vertical Japanese text was downloaded from the Tesseract GitHub repository [10] to be able to read vertical Japanese text. By default the Tesseract API only has trained weights for English text as it uses a neural network solution. Knowing this, the weights for Japanese vertical text had to be downloaded from the Github repository [10] as text in manga is almost always vertical and unlike horizontal Japanese text is read top to bottom right to left. These weights were obtained by training the optical character recognition on reading Japanese vertical text. Cropped images gathered from the object detection model were converted to solid black and white images depending on the brightness levels of the pixels in the image, this was done to lower the amount of noise the OCR had to deal with increasing signal to noise ratio.

IV. GRAMMAR REORDERING

A. Datasets

The dataset was made using a python script that would produce a dataset from an existing corpus of grammatically

correct English text. The dataset used for training the networks was a collection of movie and anime subtitles ordered into translation pairs. Only the English text was used for training the grammar correction model [28 - 30]. It was decided that this dataset should be used for the research as the research involves comics, meaning characters may speak different to how real people do such as being casual and skipping socially accepted customs such as greetings. This is also true for movies and animation and so it was decided that this may be a better choice compared to a dataset where people spoke formally and would be used for people learning a language for the first time.

This dataset was then augmented to swap the object and verb parts of the sentence to simulate text that had been translated from Japanese but not reordered. Knowing that the natural order for Japanese is SOV, the training data was split into 3 parts. With these sentences split into 3 parts, the first part at the front was kept as Subject; which is at the front of English and Japanese. The second and third parts of the English dataset were then swapped so that the English sentences in the format SVO are now in the format Subject Object Verb which would roughly be what they would be if they were translated directly from Japanese and not reordered. This allowed for the creation of a dataset that would simulate text directly translated from Japanese.

B. Training a Character Level Grammar Model

A character level model and a word level model were employed. A character model constructs sequences from characters as the smallest parts while a word level model would construct sequences from words as the smallest parts. This model uses each character as a token within the sequence to be fed into the model to produce another sequence of characters. The model allows for input sequences of up to 50 characters as the LSTM state size was set to 50 and the latent dimensionality of the encoded LSTM state was set to 256. 190000 samples had been taken from the dataset to train the character level model, split 80% for training and 20% for validation. This graph shows that the loss had levelled out and was making extremely slow progress. The loss was notable still at 0.1, 1.0 being as far away from the ground truth and 0 being the ground truth from the training data. This shows that the network was only producing output that was similar to the target data ~90% of the time. The X axis in the graph represents the number of epochs the model had been trained which is essentially time against the Y axis of loss. The network consisted of an LSTM encoder - decoder and a dense fully connected neural layer at the end. The encoder should encode the meaning and the decoder should be able to decode the meaning [17]. LSTM was used in to try and encode the natural ordering of English and try to reorder so the text is more grammatically correct.

C. Training a Word Level Grammar Model

The word level model was used with an embedding layer before the LSTM. The embedding layer was there to try and encode the meaning of the word into vector space [18]. This encoding is then passed to the LSTM to encode the meaning of the sentence. The model had been trained until the graph had

levelled out and training progress had become very slow. The model trained in 2 sessions as there is a visible anomaly in the graph, this is displayed as a spike. The model was trained on 9000 sample sentences and validated on 1000. The X axis on the graph represents the number of epochs trained against the Y axis of accuracy and loss. Despite the model being trained on the data it does not seem to be able to reorganise sentences into a more English-like grammatical sentence structure as it achieved a 0.284 score.

V. RESULTS

A. Resulting BLEU score from character and word level models

Bleu was used to test the reordering quality of both models with the character level model scoring an average of 0 and the word level model scoring 0.284 rounded to 3 decimal places. BLEU was used to compare translation quality with the official English translations on [21] which were used as the reference translation, the original Japanese version that was used it also on [21]. The character level result of 0 means that the character level model was producing output not like the reference translation. This displays an oversight of choosing to use BLEU to measure translation quality as English and Japanese are very distant languages there are often many ways to translate sentences and as such even expert human translators may have multiple different translations. Upon inspection the reordered sentences had lost their sentiment and did not contain the meaning of the reference sentence despite using some of the same words, some examples are shown in Table I.

TABLE I. EXAMPLES OF REORDERED SENTENCES

Model	Examples of reordered sentences	
	Reordered sentence	Reference sentence
Char	the name of the world we can do	the natural resources of my country
Char	but the company is the only one who can see the world	but that same year i was in london 16 years ago
Char	i just want to be a serious person in the country of the world	i just didn't realize how much it was going to cost
Word	but ive also found that im able to endure	but ive also found that im able to spend
Word	the shocking truth written in the report	the make using me check one capital feel
Word	discussed in the next staff meeting	its put up with the way

There was an inherent flaw with using BLEU for measuring translation performance as BLEU uses common words and sequences of words to provide a score between 0 and 1 of how similar 2 sentences are. It was chosen for its benefits such as being fast and easy to calculate as well as being easily understood as it has become a de facto standard. But due to the nature of Japanese to English translations there could be many different valid translations as English is more direct in terms of meaning and how things are directed compared to Japanese where a lot of meaning and direction is inferred, this means that a machine translation may have to fill in the holes resulting in a valid but completely different translation when compared to the reference translation.

B. Results from The Implementation Of The Proposed Solution

The translated image in Figure 10 using Google Translate is a very good example of the problem trying to be solved. The Object and Verb of the rightmost text box are switched and completely changes the meaning of the sentence. The text reading “It is a pity to apologize for seeing this” should read something along the lines of “It is a pity to see this apology”, with the object and very swapped the original meaning of the sentence is lost. Another problem with this image is that “man” had been translated to “adult” making the origin sentence “You are a good man too” sound strange. Moreover, faster_rcnn_inception_v2_coco trained to draw bounding boxes around Japanese text, Tesseract OCR as the optical character recognition and Google Translate as the machine translation. Finally, the result of the implementation was compared with a page from the official translation to measure translation quality using such an implementation. BLEU 2 gram produced a score of 0.069 and BLEU 1 gram produced a score of 0.159. This shows that some common words are being used but these could just be common English words.

VI. CONCLUSIONS AND FUTURE WORKS

The research aims to show a proof of concept methodology for using machine learning to translate Japanese comic books into English, while using bounding box object detection to detect text regions and experimenting with a naïve post processing grammar correction method. This implementation of the proposed methodology is functional and proves that this methodology can work in practice. Each part of the process is heavily dependent on the part before it to the point where if one part does not work well the rest cannot do perform well. However, this architecture allows modularity, where different models can be swapped in and tested. In the future work, a complete clean dataset will be created so the proposed methodology could further be implemented into a standalone application as this research this paper is just a proof of concept.

REFERENCES

- [1] Yihui He, Chenchen Zhu, Jianren Wang, Marios Savvides, Xiangyu Zhang, Bounding Box Regression with Uncertainty for Accurate Object Detection, 23 september 2018.
- [2] Smith R. 'An Overview of the Tesseract OCR Engine', Link: <https://static.googleusercontent.com/media/research.google.com/en//pubs/archive/33418.pdf>, 2007.
- [3] Wu Yonghui, Schuster Mike. Google’s neural machine translation system: Bridging the gap between human and machine translation, 2016.
- [4] 'Scanlators' freely translating 'manga,' 'anime'. The Japan Times Online. LONDON (Kyodo). 10 March 2009. (Accessed: 17 January 2020).
- [5] GitHub, manga-translator, <https://github.com/dalelyunas/manga-translator/>
- [6] Sutskever I, Vinyals O, Le Quoc V. 'Sequence to Sequence Learning with Neural Networks', Sequence to sequence learning with neural networks. In Advances in Neural Information Processing Systems, link: <http://aclweb.org/anthology/C10-1071> (), pp. 3104–3112, 2014.
- [7] Meyer, Charles F. Introducing English Linguistics International (Student ed.). Cambridge University Press, 2010.

- [8] Frey, Carl Benedikt; Osborne, Michael A. "The future of employment: How susceptible are jobs to computerisation?". Technological Forecasting and Social Change. 114: 254–280, 1 January 2017.
- [9] Negnevitsky, M. Artificial Intelligence A Guide to Intelligent Systems, Second edition edn., 2005.
- [10] Tesseract-OCR Tesseract-OCR, Github repository, Available at: <https://github.com/tesseract-ocr/tessdata/> (Accessed: [09/04/2019]).
- [11] Turing, A. M. Computing Machinery and Intelligence, 1950.
- [12] Google. Google AI Blog, Available at: <https://ai.googleblog.com/2018/05/duplex-ai-system-for-natural-conversation.html>, 2018.
- [13] Alex Krizhevsky, Ilya Sutskever, Geoffrey E. Hinton. ImageNet Classification with Deep Convolutional Neural Networks, proceedings of Advances in Neural Information Processing Systems 25 (NIPS), 2012.
- [14] Kishore Papineni, Salim Roukos, Todd Ward, Wei-Jing Zhu. 'BLEU: a Method for Automatic Evaluation of Machine Translation', Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics (ACL), Philadelphia, pp. 311-318, July 2002.
- [15] Hochreiter Sepp, Schmidhuber J. Long short-term memory, : https://www.researchgate.net/publication/13853244_Long_Short-term_Memory, 1997.
- [16] Van Horn G. , Mac Aodha O. et al. The iNaturalist Species Classification and Detection Dataset, IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), April, 2018.
- [17] Cho K. et al. Learning Phrase Representations using RNN Encoder–Decoder for Statistical Machine Translation June, 2014.
- [18] Mikolov T, Chen K , Corrado G, Dean. Efficient Estimation of Word Representations in Vector Space, Sep 2013.
- [19] Jansen. S. Word and Phrase Translation with word2vec, Link: <https://arxiv.org/pdf/1705.03127.pdf>, April 2018.
- [20] Lee Young-Suk ,Zhao Bing, Luo Xiaoqiang. 'Constituent Reordering and Syntax Models for English to Japanese Statistical Machine Translation', Proceedings of the 23rd International Conference on Computational Linguistics, link: <http://aclweb.org/anthology/C10-1071>, pp. 626–634, August, 2010.
- [21] MangaOnWeb, Online, Available at: <http://mangaonweb.com/> (Accessed: [14/04/2019]).
- [22] GitHub, pkulzc tombstone Nealwu. Tensorflow detection model zoo, Available at: https://github.com/tensorflow/models/blob/master/research/object_detection/g3doc/detection_model_zoo.md (Accessed: [04/03/2019]).
- [23] tzutalin LabelImg Github repository, Available at: <https://github.com/tzutalin/labelImg> (Accessed: [14/04/2019]).
- [24] Ko Ransom. Say Hello to Black Jack, Available at: <https://www.animenewsnetwork.com/news/2012-08-24/say-hello-to-black-jack-author-makes-title-copyright-free> (Accessed: [04/03/2019]).
- [25] Le Blanc S, Masami M. Award Winning Manga to Be Freely Used by Anyone for Anything Anytime, Author Will Not Request Royalties, Available at: <https://soraneews24.com/2012/08/22/award-winning-manga-to-be-freely-used-by-anyone-for-anything-anytime-author-will-not-request-royalties/> (Accessed: [04/03/2019]).
- [26] Olivia S. , Wired UK. MANGA ARTIST LETS ANYONE USE HIS WORK, Available at: <https://www.wired.com/2012/08/shuho-sato-royalty-free/> (Accessed: [04/03/2019]).
- [27] EdjeElectronics EdjeElectronics, Github repository, Available at: <https://github.com/EdjeElectronics/TensorFlow-Object-Detection-API-Tutorial-Train-Multiple-Objects-Windows-10> (Accessed: [08/04/2019]).
- [28] Pryzant R, Chung Y, Jurafsky D, Britz D. JESC: Japanese-English Subtitle Corpus, link: <https://web.stanford.edu>, 2018.
- [29] Sharif, M. S., Abbod, M., Al-Bayatti, A., Amira, A., Alfakeeh, A. S. and Sanghera B. "An Accurate Ensemble Classifier for Medical Volume Analysis: Phantom and Clinical PET Study," in IEEE Access, vol. 8, pp. 37482-37494, doi: 10.1109/ACCESS.2020.2975135, 2020.
- [30] Pryzant R, Chung Y, Jurafsky D, Britz D, Stanford Natural Language Processing Group JESC Japanese-English Subtitle Corpus, Available at: <https://nlp.stanford.edu/projects/jesc/> (Accessed: [14/04/2019]).