Optical character recognition and long short-term memory neural network approach for book classification by librarians

Journal of Physics: Conference Series **1567** (2020) 032034 doi:10.1088/1742-6596/1567/3/032034

Optical character recognition and long short-term memory neural network approach for book classification by librarians

YD Rosita[1*], YN Sukmaningtyas[1]
[1]Study Program of Informatics Engineering, Universitas Islam Majapahit

*Corresponding Author: yesidiahrosita@gmail.com

Abstract. The book is classified by librarians that use Decimal Dewey Classification (DDS) System. It is used for cataloging and indexing books. DDC has three divisions, a ten, a hundred, and a thousand. The book subject is reflected in each division. Commonly, to know the book content, librarians read the book title. Then, they identify the book index in DDC system. Nevertheless, it requires more time. To read the book title, Optical Character Recognition (OCR) aids them to get the book title efficiently that convert the image of the book cover into the text-editable. Librarians use a web camera to scan the book cover, especially the book title area. There are three steps for pre-processing, the lowercase changing, the useless word removing, and tokenizing. To detect the book categories, Long Short-Term Memory Neural Network is good implemented in this research. It is almost used for text classification. In this research, It gives high performance that achieves more than 92% accurately.

1. Introduction

Lately, the book categorizing with DDC is an important job for a fresh librarian in a library or other alike placesIt has a subject code and a descriptor which about 6 to 10 digits in length[1]. In the first step to knowing the DDC code's book, the librarians have to read the DDC guide that requires more time. Commonly, the DDC code is known by the book subject that can be identified by the book title. The previous research applied the neural network method that has great truthfulness for text classification. However, the neural network method also has limitations such as an architecture system for the training process, defining the window size, and feature extraction[2,3]. The limitations of text classification make authors inspired to implement Recurrent Neural Network (RNN) for text classification. To find out the DDC code precisely, several processes have to carry out before classifying a book title. They are that are transforming uppercase to lowercase, eliminating punctuation, tokenizing, stemming, and deleting words that are in the stop word list. The goal is to get prominent words and establish it simpler for the training process.

Optical Character Recognition (OCR) can classify the book subject efficiently by transforming the image into text-editable[4]. It is a useful approach to assists librarians in reporting. They do not require the book title typing. They only put the book on the scanner machine or web camera and get DDC code from text classification process.

The previous research implemented Support Vector Machine (SVM) for text classification[6]. The method achieves high performance. It is competent to hypothesize simultaneously in superior dimensional characteristic space. It also does not need characteristic selection. The method of Naive Bayes is used to present the great result. The author succeeds to prove that these methods are high performance too among other common methods.

Hassan [7] assesses their experimentation that used SVM. They enhance micro-average f-measure from 0.868 to 0.919 and macro-average f-measure from 0.865 to 0.920. Then, the result gives the level refinement 6.36 and 28.78% with SVM and Naive Bayes in succession which integrates the extracted information from Wikitology. SVM is a binary classifier approach and a new approach for classification of both nonlinear and linear. However, the approach works well with superior dimensional data. The author divided documents with a ratio [60 -40] in training and [-80 -20] in testing. The system reaches good performance which uses 80 training documents and 20 testing documents.

In this study, we proposed Optical Character Recognition (OCR) for efficient typing which only gets the book title in editable-text by scanning the book cover. The system added a web camera or scanner machine to support it. Certainly, the scanning focuses on the book title area. The aim to assure the text is the book title. Then, the system can classify text using Long Short-Term Memory (LSTM) Neural Network. The result is the main DDC code which is the first step to assist in determining DDC code completely.

2. Material and Method

2.1. Hardware and Software

To support OCR implementation, the computer hardware specification is needed (see table 1). In this study, it also added a web-camera or scanner machine to scan the book cover. The higher the hardware specification, the better the performance system.

To design the architecture system, Matlab is a recommended tool. It is a completely developed tool for research that is a multi-paradigm numerical computing environment and fourth-generation programming language[9]. It is equipped with many functions like text classification function using Long Short-Term Memory Neural Network, capturing an image, converting image into text-editable using OCR, etc.

Table 1. Hardware specification

Pheripheral	Specifications
Memory	OnBoard Memory 2 GB / 4 GB, DDR3 1333 MHz SDRAM
Processor	Intel Core i3 Processor
Storage	5400RPM, 500GB
Operating System	Windows 8
Power Adapter	Built-in Bluetooth™ V4.0
Chipset	Intel HM76/HM70 Express Chipset
Card Reader	2 -in-1 (SD/ MMC)
Display	16:9 HD (1366x768), LED Backlight, 11.6"

2.2. Collecting Data

Data are available in GitHub source that is donated by Iwana [10]. It contains 207,572 book titles with subject classes that are hosted by Amazon.com. The language of data is the English language. The proposed study used several data from Iwana's data and added it from other sources like the Online Computer Library Center (OCLC) web service. The election of texts that used for the training process can also influence the accuracy level of classification such as the similarity of words that have different meanings and classes. The data have 3000 items for the training dataset and 300 items for the testing dataset. The data contains the book titles and their classes. For the testing process, the book cover image is added that is provided by Amazon.com. Although the DDC code has three divisions that are tents part, hundreds part, and thousands part, the researchers used three classes in tens part because it requires high specification hardware when training with large amounts of data. The tens part is known as the main division. Table 2 explains the dataset used. The three classes are widely found in several libraries that are language, technology, and history.

Table 2. Dataset

DDC Code	Class	Number of Data	
		Testing dataset	Training dataset
400	Language	100	1000
600	Technology	100	1000
900	History	100	1000

A lot of data that supports the establishment of a classification model to obtain strong accuracy. On the other hand, it does not only hinge on the number of data but also the pattern of the system architecture is reasonable.

2.3. System Architecture

The mechanism of text reading using Optical Character Recognition is shown in figure 1. The web-camera captures the book cover to get the text, especially the book title area. Furthermore, the text will be pre-processed to get basic words in segmentation (see figure 2). The more obvious the title of the book, the better the text reading. It has to ensure that the written is clear because it takes effect the result as input for the next process, text classification.

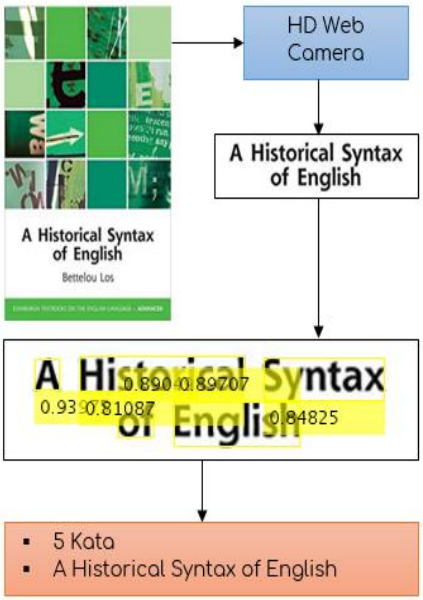

Figure 1. OCR procedure

Figure 2 shows the basic procedure which is used in this research. It is separated into two sections that are the training process and the testing process. A book cover, especially the title area, is the main input. A camera captures the book cover. Then, OCR will transform the image of the book cover into the text-editable. Furthermore, the system gains the important words. This pre-processing is needed for standardizing the words (see figure 2).

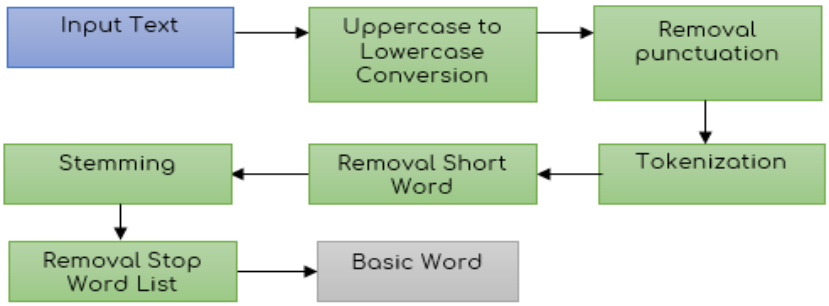

Figure 2. Pre-processing procedure

The method consists of four stages to gain pure words. For instance, there is the text that contains "A Historical Syntax of English". The system has to get words "historic syntax english".

- Transforming uppercase to lowercase. It converts to the capital letter into the lower letter. For instance, the text of "A Historical Syntax of English" is " a historical syntax of english".
- Eliminating punctuation. It cuts out the punctuation marks, such as *,&,^,#, etc.
- Tokenizing. It separates the text into tokens. For instance, "a historical syntax of english" has five tokens that are "a", "english", "historical", "syntax", and "of".
- Eliminating the short words. It cuts out the short words or its length less than equal to the determined number. For instance, "a", "in", "at", "for", "from", etc.
- Stemming. It cuts out the prefix and the suffix words to get the essential words. For instance, "important" becomes "import".
- Deleting word that are in stop word list. It cuts out unimportant words which include the dataset. For instance, "based on", "however", "several", etc.

Table 3. LSTM neural network architecture

The Layer Name	Descriptions
Sequence Input	One dimensions
Learning Rate	0.01
Word Embedding	A hundred dimensions and 2573 exclusive words
LSTM	A hundred and eighty hidden units
Softmax	Softmax
Fully-connected	Thirty nine fully connected layer
Classification Output	Crossentropyex

To create a classifier model, we proposed the LSTM Neural Network. It is a subcategory of Recurrent Neural Networks (RNNs) to decrease the charge of the computation unit[11], [12]. The first step is word encodings that reorganize the training data into sequences of numeric indices. To enter sequence data into LSTM neural network, the system utilizes sequence input and sets the input size to one. Furthermore, it is processed into the word embedding layer that is in dimension 100 and the word encoding. The author set 180 hidden units in LSTM layer and output mode to 'last' for a sequence-to-label classification problem. Lastly, setting a fully-connected is required that has the same size as the number of classes, a softmax layer, and a classification layer.

The system only calls the classifier model which is built for the testing process. The outcome of catch the image of book cover by OCR will be transformed from the image-based into text-based. [13]–[15]. The words from the text are important things for pre-processing input because the classifier can forecast the text category.

3. Results

There are two sections to describe the experimental result. In the first section, we applied OCR to get editable-text as a book title from a book cover. It only captures the book title area not the whole area of the book cover. Its aim is assuring that its catching is the book title. The system can count the number of words in the book title and exposes the words. When the catching is unclear, the system will be horrible. Unclear terms included polluted text and freestyle font type. Nevertheless, the segment words can catch but not know the text. If the book title consists of freestyle font type, the system can not designate certainly. It usually displays some punctuations or symbols like “2?o/5//¢/'oa(fax(9/ of Zf.'s=4fi/?”.

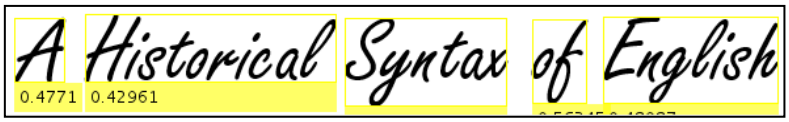

Figure 3. Freestyle font type

The training process needs much time to obtain a greater truthfulness. This achievement was obtained at 630th iteration in the 30th epoch. Figure 4 presents that almost no data was disappearance. The greatest accuracy reaches 92.33%. This proves that the system can classify texts precisely. However, the system does not do well when finding similarities in data. For instance, the text "Optimal route for intelligence traveling" is a technology class in the dataset but the system recognizes it by reading per word to derive features. It contains "route", "travel" in lowercase that are the history class. The system gives code 900 that informs the book is including history class (see table 2).

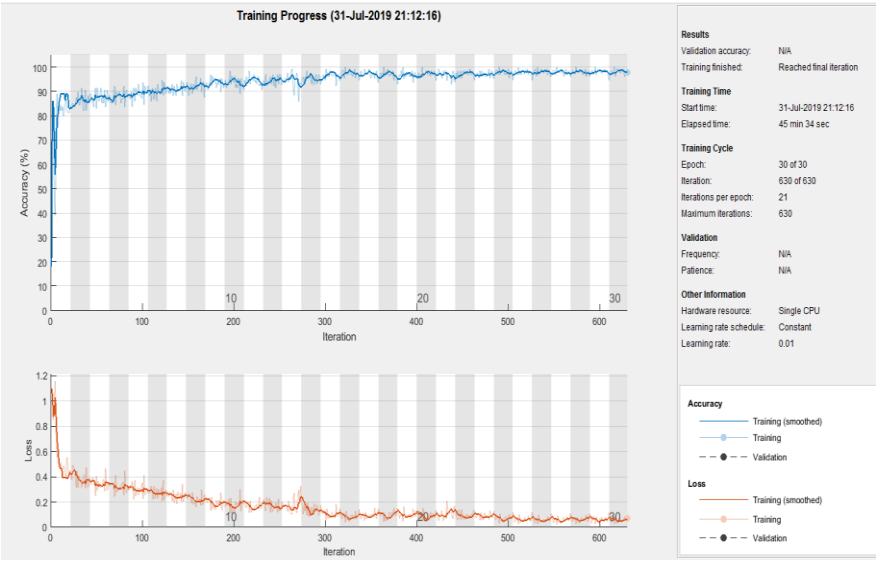

Figure 4. Training process

Figure 4 displays a system performance with LSTM architecture that uses 3000 data in 3 classes is an almost perfect achievement. It needs 45 minutes and 34 seconds to get greater accuracy and minor failure in the training process. The accuracy is reached at 630th iteration in 30th epochs. It is used for building a classifier model. The dataset accommodates the book title and class that used for as main division code. The classifier model cultivates input to produce the main division code that can help librarians for labeling (see Table 2).

4. Discussion

In this study, it has been proven that the proposed system can be used well even though it includes deficiencies. When the system is applied to a smartphone is better and efficient. For future work, the

system is not only catalogs by a title but also a synopsis. So, the system can deliver a detailed DDC code.

5. Conclusion

Based on the experiments, the collaboration of OCR and LSTM Neural Network can work well. OCR and LSTM neural networks can assist in getting DDC code that is an efficient book classification way. Based on the experiments, the collaboration of OCR and LSTM Neural Network can work well. OCR and LSTM neural networks can assist in getting DDC code that is an efficient book classification ways. Exactly, they are useful to help the librarians. OCR reconstructs the book title in the image into an editable-text that is a beneficial way to trim typing time for documentation by librarians. To get DDC code, the book title in editable-text will be categorized by LSTM neural network. The system can not work well when the book title is not clear such as freestyle font type, polluted, inadequate amount of data, and not perfect system architecture.

Acknowledgment

Many thanks for research funding to the Directorate General of Higher Education and the Ministry of Research and Technology, Indonesia.

References

[1] Zuccala A and Garcia NR 2018 Reviewing, indicating, and counting books for modern research evaluation systems [Forthcoming] *Springer Handbook of Science and Technology Indicators* (Switzerland: Springer International Publishing)
[2] Kamran K, Meimandi K J, Heidarysafa M, Mendu S, Barnes L and Brown D 2019 *Information* **10** 1
[3] Ren F and Deng J 2018 *Appl. Sci.* **8** 2472
[4] Alotaibi F, Abdullah MT, Abdullah RBH, Rahmat RWBOK, Hashem IAT and Sangaiah AK 2017 IEEE Access **6** 554
[5] Patel CI, Patel AP and Patel D 2012 *Int. J. Comput. Appl.* **55** 50
[6] Wei L, Wei B and Wang B 2012 *J. Softw. Eng. Appl.* **05** 55
[7] Hassan S, Rafi M, and Shaikh MS 2011 Comparing SVM and Naïve Bayes classifiers for text categorization with Wikitology as knowledge enrichment Proc. 14th IEEE Int. Multitopic Conf. 2011, INMIC 2011, pp. 31–34
[8] Fatima S 2017 *Int. Res. J. Eng. Technol.* **4** 141
[9] Chen PY 1984 Research Methodology pp. 36–40.
[10] Iwana BK, Rizvi STR, Ahmed S, Dengel A and Uchida S 2017 Judging a Book by its Cover *Comput. Vis. Pattern Recognit 1610.09204v3*
[11] Sen S and Raghunathan A 2019 *IEEE Trans. Comput. Des. Integr. Circuits Syst.* **37** 2266
[12] Khalil K, Eldash O, Kumar A and Bayoumi M 2019 *IEEE Trans. Circuits Syst. II Express Briefs* 66 1888
[13] Vijayarani S and Sakila A 2017 *Int. J. Adv. Res. Comput. Commun. Eng.* **6** 55
[14] Isheawy NAM and Hasan H 2015 *J. Comput. Eng.* **17** 2278
[15] Jiang Y, Dong H, and Saddik AE 2018 *IEEE Access* **6** 60128