Correction of Annotation Affected Words in Document Images

Ritika Kumari(Reg. No. 267) Pulkit Mishra(Reg. No. 258) Jyotika Yadav(Reg. No. 238)

Under the supervision of

Dr. Sanjoy Pratihar

Indian Institute of Information Technology Kalyani

Department of Computer Science and Engineering

ABSTRACT

The aim of this project is to suggest an robust, end-to-end method to correct the annotation affected document images. We propose the solution to this problem in two parts: (1) Pre-processing the OCR input: Detection and removal of annotation that may appear in any part of the document, without causing any loss of original printed information. The proposed technique is based on localizing annotations using the area of connected components, separating annotations from letters using undirected path opening morphological operation and then regenerating the lost text using inpainting. (2) Post-processing the OCR output: Context-based error correction for correcting OCR cognitive, non-word, real-word and short form errors. This technique is based on sequence to sequence architecture with bidirectional RNN in encoding layer and Bahdanau Attention in decoding layer for spelling correction.

1. INTRODUCTION

The dawn of the digital age on mankind has laid the foundation of connectivity, fostering access and exchange of information practically anywhere in the world. With the growth of digitization and getting information online, there is a need to archive physical books into digital ones. This will make books and documents searchable and usable by the users. Information can be present in any form, the most common being textual and graphical documents. Capturing and curating documents such as magazines, newspapers, journals and scientific articles is the primary requirement for a digitized, inter-connected society. While most of the textual documents can be stored with a decipherable textual component, the story is not the same for annotation affected

documents where annotation can be hand drawn marks such as ticks, crosses, circles made on these documents. In the attempt to digitize books and documents one needs to identify the characters and words of the documents. This is done using OCR. Optical Character Recognition (OCR) is the process of identifying typed, handwritten or printed textual characters within a document containing scanned images or photographs with the help of various image processing and pattern recognition techniques. The text obtained by OCR systems often suffers from low accuracy owing to irregularities in images, poor scans or simply the nature of arrangement of letters in a word. These errors severely hamper the quality and readability of a converted document. Identifying and rectifying such erroneous characters in every OCR-processed document manually is a tedious task due to the sheer volume of data. Consequently, a methodology is required to identify such OCR errors and rectify them in order to enforce standards of purity and quality of the archived data. Among many of the shortcomings of OCR while reading a document image, we are addressing a very important one : if the documents have been annotated. If the document is annotated with some mark (a tick, a cross, a circle - perfect or imperfect) then the OCR is unable to perform, the outcome is not satisfactory and the accuracy is also hampered. The aim of this project is to identify and correct the document image affected by such annotations.

To tackle this problem, we propose a method which consists of two parts: (1) Pre-processing : Detection and removal of annotation that may appear in any part of the document, without causing any loss of original printed information. To achieve detection, we propose a solution based on connected component analysis and path opening morphological operation. Then inpainting is used to regenerate the damaged text due to removal of annotations.

(2) Post-processing: Context-based error correction for correcting OCR cognitive, non-word, real-word and short form errors. This technique is based on sequence to sequence architecture with bidirectional RNN in encoding layer and Bahdanau Attention in decoding layer for spelling correction.

The rest of the report is organized as follows: Section 2 talks about the related works done on the problem. Section 3 discusses our approach in detail. Section 4 contains the results. Section 5 includes the conclusion and Section 6 includes the references.

2. RELATED WORK

2.1. Detection and Removal of Annotation

Rovina and Seema (2016) proposed a solution to remove the hand-drawn underlines and annotation lines from documents like circular annotation, strikethrough lines, straight underlines, touched and untouched underlines, broken underlines and other text surrounding lines. An RGB image is converted into lab color space for global features then divided into different clusters so that annotated areas should be extracted from the background. Novelty of this method lies in its ability to compute the annotation area whether it touches the word and also detect strikethrough lines. Inpainting is used for removing lines and filling the lost parts [1].

Pratihar et al. (2012) introduced an algorithm for detection and removal of hand drawn underlines present in a scanned document. In this paper detection of underlines and detection of edges of their covers was done. This algorithm worked for all types of underlines whether

touched or untouched and whether lines are curved and bent, as commonly seen when drawn by hand [2].

Adak and Bidyut (2014) proposed a scheme to detect strike-through text/words. Graph based model was used to represent a textual connected component as a graph. This approach dealt with strike-through text in handwritten documents. This graph based algorithm was tested on English, Bengali and Devanagari scripts. This approach can be extended to some other scripts if basic structural features of scripts are known [3].

Bai et al. (2004) proposed a three-module approach for underline detection and removal in Chinese/English OCR. In this paper the detection module used two methods for detection of lines connected component analysis and bottom edge analysis. Connected component analysis was for detecting touched underline and doubtful underline. This approach can deal with untouched, touched, broken and curved underlines [4].

Das and Banerjee (2014) presented a technique that has been used for underline detection and removal in a Bengali and English document which has been confirmed to work for any kind of underline like touched, untouched using Gabor filter and connected component analysis. In this paper the first document page was taken as input then the binarization algorithm was used. After that Gabor filter was applied on the binarized image. This helps in distinguishing headline region from the underline region. In the underline detection module, first Gabor filter is applied in a specific direction to detect underline region and then connected component analysis is used to detect particular underline. In another part i.e. underline removal module uses nearest neighbor approach. This algorithm is used for both touched and untouched underline. [5].

Oba et al. (2009) introduced an underline removal method specific to Japanese business documents. Firstly it deals with multi resolution images and normalizes the input image. Secondly it reduces processing time and thirdly it selects various underlines. After that finally, it removes table ruled lines. Line template matching was used for detection in business documents and line template matching can also remove both thin and thick underlines. This method can remove touched, untouched and table ruled lines. [6].

Pinto et al. (2004) introduced the problem of handwritten underline removal. The use of these underlines are very much found in books and removals of these underlines are the national goal of the national libraries in their process of building digital libraries. Firstly the binarization of image is done then after underline removal was used for character recognition. Mathematical morphology method was used and line detection was based on small eigenvalues. Removing these lines is important for good OCR performance and for visual appearance of degraded documents[7].

Saba et al. (2014) introduced a comparison of various preprocessing techniques in offline script recognition. Some of them are hard such as line removal from text documents, skew removal, reference line detection etc. [8].

Singh and Mahajan (2014) introduced Fuzzy c means as an unsupervised clustering algorithm in which one element belongs to one or more clusters. Fuzzy c-means is advanced from k-means clustering and it did better work [9]

2.2. Post-OCR Correction

Evans and Zhai proposed a context-sensitive word-error correction system which is based on confusion mapping that uses confusion probabilities in order to identify sequences that

are frequently and convert them into the most probable correct sequence. In simpler terms, it models how likely one letter has been misinterpreted as another [10].

Atwell and Elliittm proposed to apply part-of-speech (POS) tagger and the grammatical rules of the English language in order to capture real-word errors in the OCR text. There are several rules in the English language such as a verb can be followed by gerund object but it cannot be followed by a second verb, while another rule states that a third person verb in the present tense must always take an “s”. The aggregate of all such rules drives the logic of the algorithm and helps in achieving a reasonable context-based OCR error correction scheme [11].

MaysE and Damerau used word trigrams to capture and correct non-word and real-word errors. The idea is to use a combination of lookup dictionary to correct non-word errors,and a statistical model to correct real-word errors according to their context [12].

GoldingR and A Bayesian proposed a Bayesian Classifier that treats the real-word errors as ambiguous, and then tries to find the actual target word by calculating the most likely candidate based on probabilistic relationships between the error and the candidate word [13].

Golding R.A and SchabesY joined a lot of the previous ideas into a concrete solution;it is a POStagger enhanced by word trigram model and a statistical Bayesian classifier developed to correct real-word errors in OCR text. Overall,the mixture of these techniques hugely improved the OCR post-processing error correction rate [14].

Hisao Niwa and Kazuhiro Kayashima proposed an OCR post error correction method based on patterns learning,wherein the list of correction candidates is first generated from a lexicon, then the most proper candidate is selected as a correction based on the vocabulary and grammar characteristics surrounding the error word [15].

Lon-Mu Liu and Yair M. Babad proposed a statistical method for auto-correction of OCR errors; this approach uses a dictionary to generate a list of correction candidates based on the n-gram model. Then, all words in the OCR text are grouped into a frequency matrix that identifies the existing sequence of characters and their count. The correction candidate having the highest count in the frequency matrix is then selected to substitute the error word [16].

Kazem Taghva and Julie Borsack proposed an improved design of clustering technique to build a set of groups containing all correction candidates. Then, several iterations of word frequency analysis are executed on these clusters to eliminate the unlikely candidate words. In due course, only a single candidate will survive to replace the misspelled word [17] .

Michael L. Wick and Michael G.Ross proposed the use of a topic model to correct the OCR outputtext. It is a global word probability model,in which documents are labeled with a semantic topic having a specific independent vocabulary distribution. In other words, every scanned document is semantically classified according to its topic using an unsupervised training model. Every misspelled word is then corrected by selecting the correction candidate that belongs to the same class of the actual error [18].

K. Kise and T. Shiraishi proposed a divergent approach based on syntactic and semantic correction of OCR errors; the idea pivots around the analysis of sentences to deduce whether or not they are syntactically and semantically correct. If a suspicious sentence is encountered, possible correction candidates are generated from a dictionary and grouped top-down with respect to their strongest syntactic and semantic constraints. In the long run, the candidate on the top of each group is the one that substitutes the corresponding OCR error [19].

3. PROPOSED METHODOLOGY

The end-to-end method that we propose in this paper with an aim of correcting errors in annotation affected documents images comes as a combination of two modules. The first module is a pre-processing module that runs before the image is fed into the OCR and is described in section 3.1 . The second module is a post-processing module that is run on the output generated from the OCR and is described in section 3.2 .

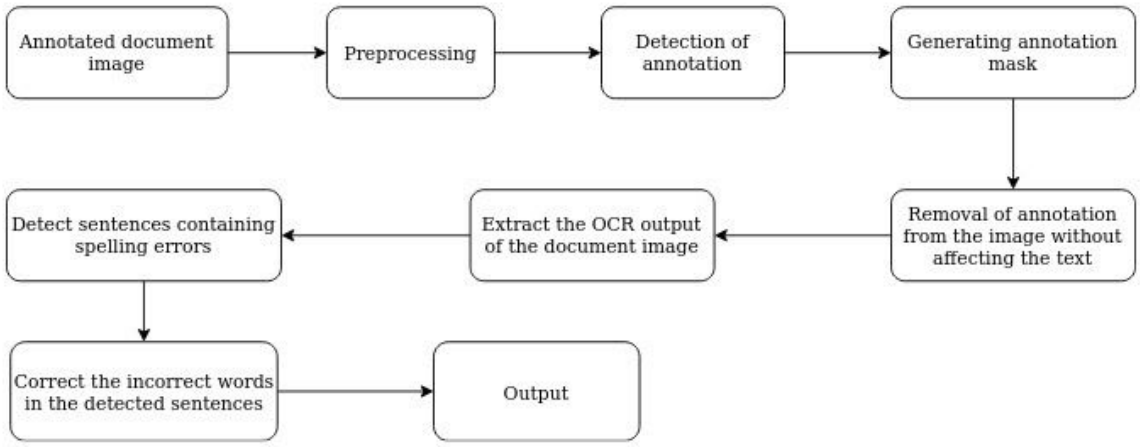
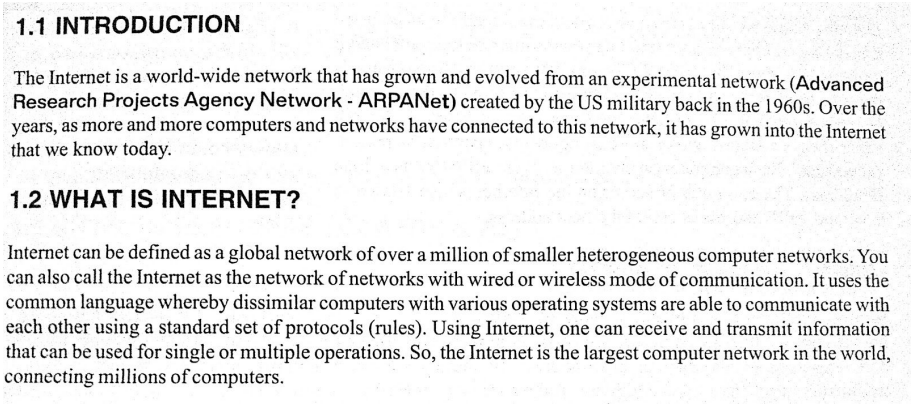


Fig. 1 Workflow of the entire process

Throughout the paper we will follow the given sample image to understand the working and results at each step of the process:



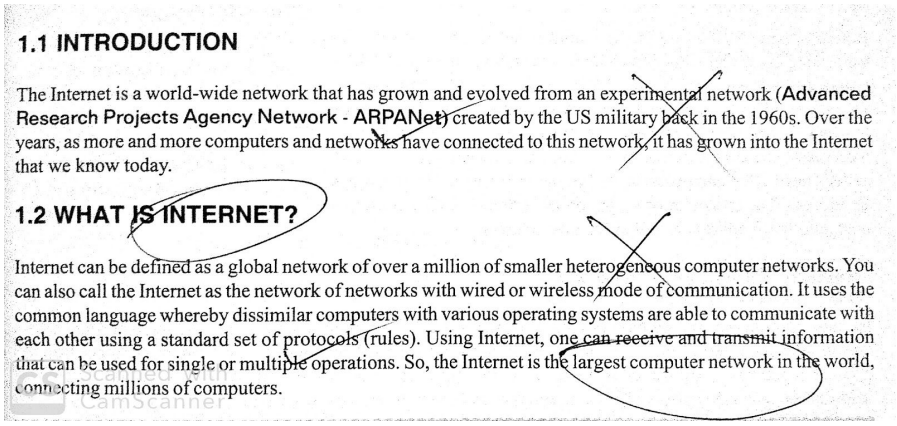
1.1 INTRODUCTION

The Internet is a world-wide network that has grown and evolved from an experimental network (**Advanced Research Projects Agency Network - ARPANet**) created by the US military back in the 1960s. Over the years, as more and more computers and networks have connected to this network, it has grown into the Internet that we know today.

1.2 WHAT IS INTERNET?

Internet can be defined as a global network of over a million of smaller heterogeneous computer networks. You can also call the Internet as the network of networks with wired or wireless mode of communication. It uses the common language whereby dissimilar computers with various operating systems are able to communicate with each other using a standard set of protocols (rules). Using Internet, one can receive and transmit information that can be used for single or multiple operations. So, the Internet is the largest computer network in the world, connecting millions of computers.

Fig.2 Original Sample Document Image



1.1 INTRODUCTION

The Internet is a world-wide network that has grown and evolved from an experimental network (**Advanced Research Projects Agency Network - ARPANet**) created by the US military back in the 1960s. Over the years, as more and more computers and networks have connected to this network, it has grown into the Internet that we know today.

1.2 WHAT IS INTERNET?

Internet can be defined as a global network of over a million of smaller heterogeneous computer networks. You can also call the Internet as the network of networks with wired or wireless mode of communication. It uses the common language whereby dissimilar computers with various operating systems are able to communicate with each other using a standard set of protocols (rules). Using Internet, one can receive and transmit information that can be used for single or multiple operations. So, the Internet is the largest computer network in the world, connecting millions of computers.

Fig.3 Document Image with Annotations

3.1. Detection and Removal of Annotation

The detection and removal of annotation serves as a necessary preprocessing step that helps in feeding the OCR with cleaner images as input, which as a result decreases the number of errors made by the OCR in reading the characters and thereby facilitating a better intermediate result for the NLP based post processing to work on.

Input used for the preprocessing part of the algorithm is a scanned image with text that contains hand drawn annotations, which are essentially marks such as imperfect ticks, crosses and circles.

The steps involved in the algorithm of the proposed work are as follows :

1) **Preprocessing of the image**
    a) Correction of skewed images
    b) Set DPI of the image (300, 300)
    c) Apply adaptive mean threshold along with opening and closing morphological operations
    d) Smooth the image using gaussian blur and perform bit_or operation on thresholded image and blurred image.
2) **Detection of annotations present in the image**
    a) Get all the connected components from the image
    b) Calculate the mode of the areas of the connected components. Initially, We considered the height of the connected components for filtering annotations from original text and took mean as the metric of measurement. After some experimentation we came to the conclusion that area would work best along with mode as the metric of measurement.
    c) Keep all those connected components which have an area greater than five times the mode and remove all other components. We considered 5 as the threshold value after observing the pattern on a very large dataset of annotated documents.

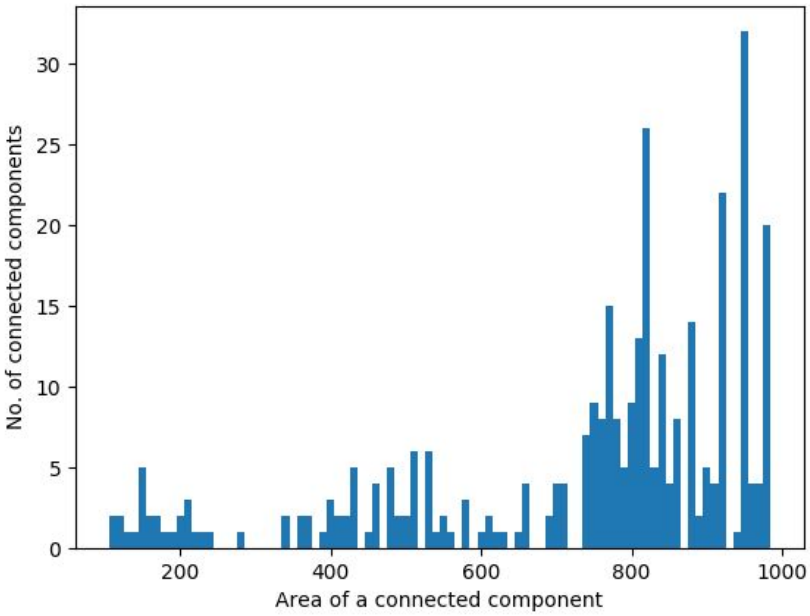


Fig. 4 Graph between area and no. of connected components

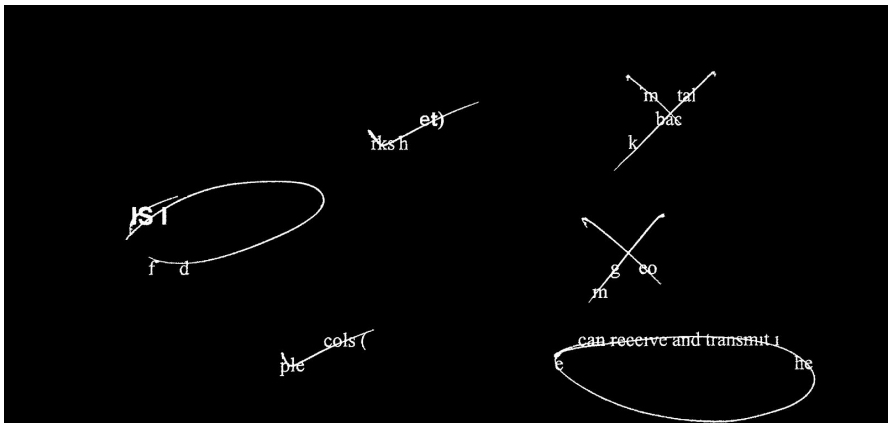


Fig. 5 The components of the document image containing annotations

3) Creation of annotation masks

This step is done using path opening and closing morphological operations.

The traditional morphological filter used for the analysis of linear structures in images is the union of openings(or the intersection of closings) by linear segments. However structures in images are rarely strictly straight, and as a result a more flexible approach is needed. An extension to the idea of using straight line segments as structuring elements is to use constrained paths, i.e.discrete, one-pixel thick successions of pixels oriented in a particular direction, but in general forming curved lines rather than perfectly straight lines.

However, the number of such paths is prohibitive and the resulting algorithm by simple composition is inefficient.

Since the annotations that are hand drawn on the document images are not straight lines, but also have circular markings that's why we implemented mask creation using path opening. Path openings and closings have flexible line segments as structuring elements. These line segments have the ability to adapt to local image structures, and can be used to detect lines that are not perfectly straight. They also are a convenient and efficient alternative to straight line segments as structuring elements when the exact orientation of lines in the image is not known. These path operations are defined by an adjacency relation, which typically allows for lines that are approximately horizontal, vertical or diagonal. However, because this definition allows zig-zag lines, diagonal paths can be much shorter than the corresponding horizontal or vertical paths. This undoubtedly causes problems when attempting to use path operations for length measurements. There are two ways in which one can perform path opening, 1. Directed Path Opening, and 2. Undirected Path Opening. Directed path opening is a morphological opening over all possible paths of a specific length and general direction. A path direction represents a 90 degree cone within which paths are generated. The paths are formed by single pixel steps in one of three

directions (in 2D): the main direction, or 45 degrees to the left or right. While in undirected path opening, opening is done in all possible directions and of all possible lengths. Here, to achieve the desired mask we have used undirected path opening in constrained mode, which means, the path construction described above is modified such that, after every alternate step, a step in the main direction must be taken. This constraint avoids a zig-zag line that causes the path opening to yield much shorter lines for the diagonal directions if the lines in the image are thicker than one pixel. It also reduces the cone size from 90 degrees to 45 degrees, making the algorithm more directionally-selective.

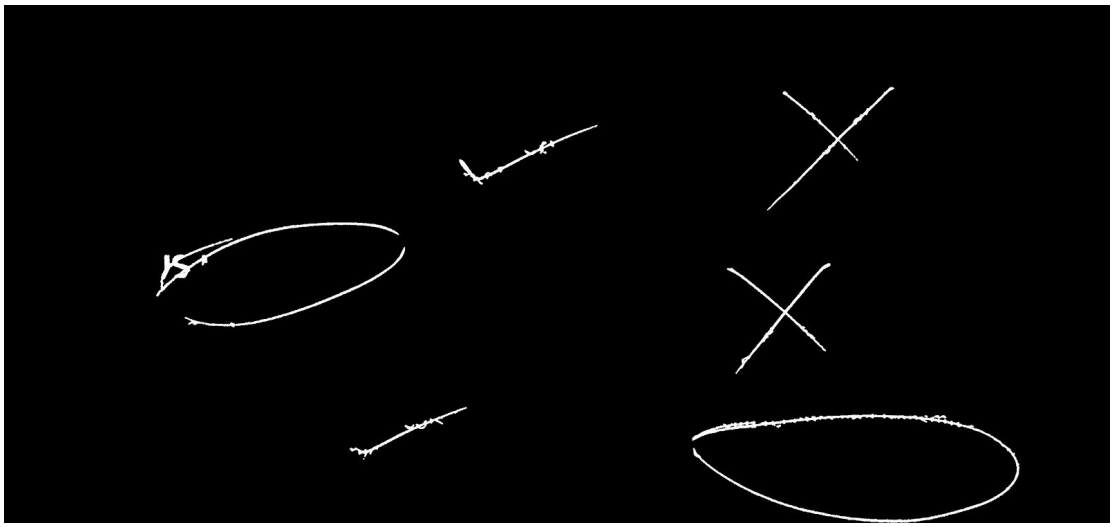


Fig. 6 Annotation masks generated by undirected path opening

4) Inpainting to regenerate affected text

This mask thus generated using undirected path opening is used for inpainting the annotation affected text area. There are many methods for performing inpainting on damaged images. On trial of such methods we found the one most suitable to solve our problem. This algorithm is

based on the paper “An Image Inpainting Technique Based on the Fast Marching Method” by Alexandru Telea in 2004. It is based on the Fast Marching Method.[23] Considering a region in the image to be inpainted, this algorithm starts from the boundary of this region and goes inside the region gradually filling everything in the boundary first. It takes a small neighbourhood around the pixel on the neighbourhood to be inpainted. This pixel is replaced by normalized weighted sum of all the known pixels in the neighbourhood. Selection of the weights is an important matter here. More weightage is given to those pixels lying near to the point, near to the normal of the boundary and those lying on the boundary contours. Once a pixel is inpainted, it moves to next nearest pixel using Fast Marching Method. FMM ensures those pixels near the known pixels are inpainted first, so that it just works like a manual heuristic operation.

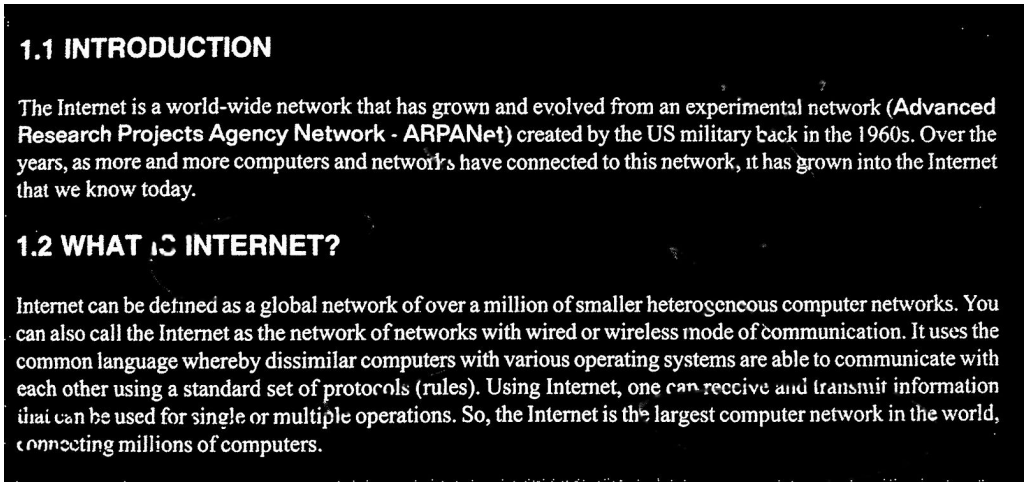


Fig.7 Resulting document image after Inpainting

3.2. Post-OCR Correction

Even after removal of annotation from the document images, the OCR output is not precise. To tackle this problem, we proposed a post-processing technique using Natural Language

Processing and Deep Neural Networks. The solution is divided into two main parts : (1) Dictionary based detection of incorrect words and (2) Context based correction of incorrect words.

3.2.1 Dictionary Based Error Detection

In order to correct the errors arising due to annotation, it was necessary to identify if any error is present at all in the first place. This was achieved by a simple and yet powerful python library called pyenchant. Pyenchant is essentially a set of python bindings of the Enchant library and with its help we were able to identify if a word is present in the dictionary or not. If the word is not present in the dictionary then it gets marked as an error. It gave excellent results for words present in the English dictionary. However, for numbers and proper nouns it became necessary to make modifications to the results obtained from pyenchant. After making the necessary modifications the words with error were replaced with a blank and the word itself was stored to be used later in order to calculate the levenshtein distance.

3.2.2 Context Based Error Correction

Spelling Errors, most of the time for spelling correctness checked in the context of surrounding words. An auto corrector usually goes a step further and automatically picks the most likely word. In case of the correct word already having been typed, the same is retained.
There are different types of spelling errors. We will classify them a bit formally as,

i. Cognitive Errors: In this type of error the words like piece-peace knight-night, steal-steel are homophones (sound the same). So you are not sure which one is which.

ii. Real Word Errors: Sometimes instead of creating a non-word, you end up creating a real word, but one you didn't intend. E.g, typing buckled when you meant bucked. Or if you type in three when you meant there.

iii. Non-word Errors: This is the most common type of error like if we type langage when you meant language; or hurry when you meant hurry.

iv. Short forms/Slang: In this case may be u r just being kewl. Or you are trying hard to fit in everything within a text message or a tweet and must commit a spelling sin. We mention them here for the sake of completeness.

For automatic spelling correction of long multi-word text, the algorithm itself has to make an educated choice. The NLP algorithm is based on the machine learning algorithm. Instead of hand-coding large sets of rules, NLP can rely on machine learning to automatically learn these rules by analyzing a set of examples and making statistical inference. We used the Seq2Seq architecture with Bi-direction RNN in the encoding layer and Bahdanau Attention in the decoding layer. We also used a grid search to find the optimal architecture and hyperparameter value.

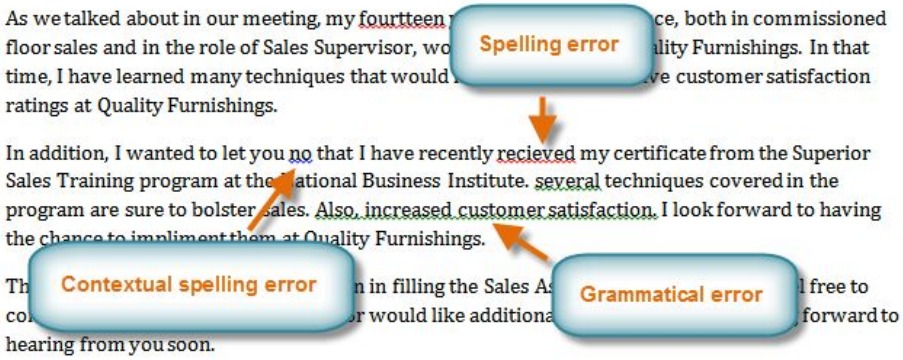


Fig. 8 Types of error in textual data

Preparation of Dataset

We used the dataset having hundred popular books from Project Gutenberg. After that the dataset is cleaned using regex expressions and sentences are extracted. Then the sentences are passed to the noise maker module. The edit operations applied for this purpose are: (i) insert characters into a sentence; (ii) delete characters from a sentence and (iii) replace a character of a sentence by another character. We used character level encoding as it can predict out of vocabulary words also.

Model Architecture

Seq2Seq - Sequence-to-sequence [21] model was first proposed in machine translation. The idea was to translate one sequence to another sequence through an encoder-decoder neural architecture. We use the attention based approach as it provides an effective methodology to perform sequence-to-sequence (seq2seq) training. We have encoded a neural network which encodes the input sequence into a vector which has a fixed length and decoder neural network will generate each of words in the output sequence in turn, which is based on vector c and previously predicted words until it meets the word ending the sentence. In the seq2seq model, we can use different network architectures for encoder and decoder networks such as RNN or *convolutional* neural networks.

The basic seq2seq model has the disadvantage of requiring the RNN decoder to use the entire encoding information from the input sequence whether the sequence is long or short. Secondly, the RNN encoder needs to encode the input sequence into a single vector which has a fixed

length. This constraint is not really effective because, in fact, word generation at a time step in the output sequence sometimes depends more on certain components in the input sequence. For example, when translating a sentence from one language into another, we are more concerned about the context surrounding the current word compared to the other words in the sentence. The *attention* technique is given to solve that problem.

Bidirectional RNN - In bidirectional recurrent neural network (BRNN) [22] that can be trained using all available input information in the past and future of a specific time frame. It contains two hidden layers of opposite directions to the same output. The principle of BRNN is to split the neurons of a regular RNN into two directions, one for positive time direction (forward states), and another for negative time direction (backward states). Those two states' output is not connected to inputs of the opposite direction states. The general structure of RNN and BRNN can be depicted in the right diagram. By using two-time directions, input information from the past and future of the current time frame can be used, unlike standard RNN which requires the delays for including future information.

The structure of BRNN is an idea is to split the state neurons of a regular RNN in a part that is responsible for the positive time direction (forward states) and a part for the negative time direction (backward states). Outputs from forwarding states are not connected to inputs of backward states and vice versa. The BRNN can principally be trained with the same algorithms as a regular unidirectional RNN because there are no interactions between the two types of state neurons and, therefore, can be unfolded into a general feed-forward network.

In training of BRNN, It can principally be trained with the same algorithms as a regular unidirectional RNN because there are no interactions between the two types of state neurons and, therefore, can be unfolded into a general feed-forward network. However, if, for example, any form of backpropagation through time (BPTT) is used, the forward and backward pass procedure is slightly more complicated because the update of state and output neurons can no longer be done one at a time.

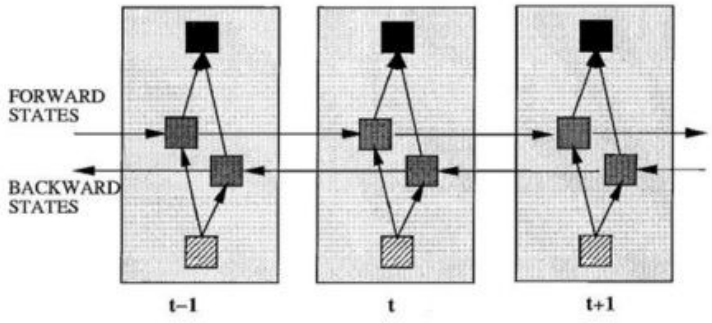
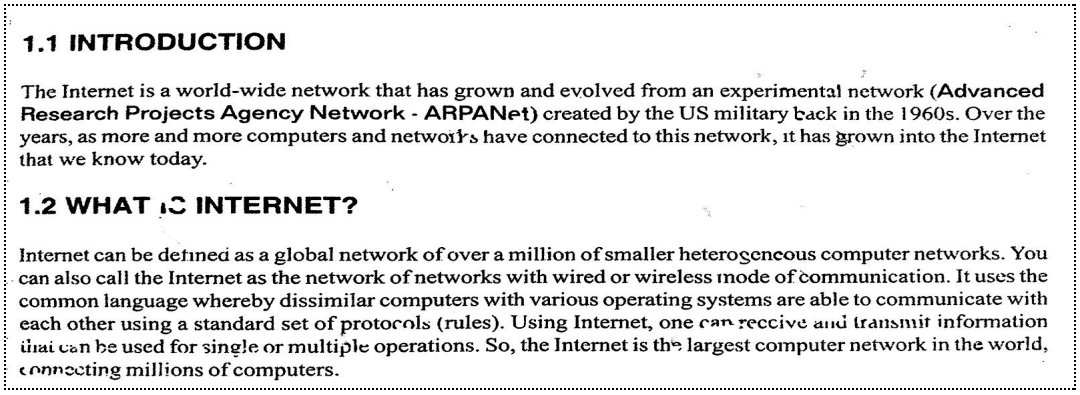


Fig. 9 Bi-Direction RNN

Bahdanau Attention - In the regular seq2seq model, we embed our input sequence into a context vector, which is then used to make predictions. In the attention variant, the context vector is replaced by a customized context for the hidden decoder vector. The result is the sum over contribution over all of the input hidden vectors. Attention [24] is important for the model to generalize well to test data, in that our model might learn to minimize the cost function during train time, but it is only when it learns attention that we know that it has an idea that it knows exactly where to look (and put that knowledge into the context) for it to generalize well to test data.

The Bahdanau attention model uses the concatenation of the forward and backward source hidden states in the bi-directional encoder and target hidden states in their non-stacking unidirectional decoder. Bahdanau has only a concat score alignment model.



1.1 INTRODUCTION

The Internet is a world-wide network that has grown and evolved from an experimental network (**Advanced Research Projects Agency Network - ARPANet)** created by the US military back in the 1960s. Over the years, as more and more computers and networks have connected to this network, it has grown into the Internet that we know today.

1.2 WHAT IS INTERNET?

Internet can be defined as a global network of over a million of smaller heterogeneous computer networks. You can also call the Internet as the network of networks with wired or wireless mode of communication. It uses the common language whereby dissimilar computers with various operating systems are able to communicate with each other using a standard set of protocols (rules). Using Internet, one can receive and transmit information that can be used for single or multiple operations. So, the Internet is the largest computer network in the world, connecting millions of computers.

Fig. 10 Image after removing annotation

```
1.1 INTRODUCTION



The Internet is a world-wide network that has grown and evolved from an experitnentaf network (Advanced
Research Projects Agency Network - ARPANSH created by the US military in the 1960s. Over the
years, as more and more computers and netwo ave connected to this networkét has Brown into the Internet
that we know today.

1.2 WHAT IS INTERNET?
Internet can be defined as a global network of over a million of smaller heterogeneous computer networks. You
can also call the Internet as the network of networks with wired or wireless mode of communication. It uses the
common language whereby dissimilar computers with various operating systems are eable to communicate with

each other using a standard set of proto rules). Using Internet, one.caz mit information
inai can he used for single or multiple operations. So, the Internet is the largest computer network int |

connecting millions of computers.
```

Fig.11 OCR output before post-processing

```
1.1 INTRODUCTION

The Internet is a world-wide network that has grown and evolved from an experimental network (Advanced
Research Projects Agency Network - Arcane) created by the US military back in the 1960s. Over the
years, as more and more computers and networks have connected to this network, it has brown into the Internet
that we know today.

1.2 WHAT IS INTERNET?

Internet can be defined as a global network of over a million of smaller heterogeneous computer networks. You
can also call the Internet as the network of networks with wired or wireless mode of communication. It uses the
common language whereby dissimilar computers with various operating systems are able to communicate with
each other using a standard set of protocols (rules). Using Internet, one can receive and transmit information
that can he used for single or multiple operations. So, the Internet is the largest computer network in the world,
connecting millions of computers.
```

Fig. 12 OCR output after post-processing

4. RESULTS

We annotated 36 pages from 3 different books:

- Hitchhiker's Guide to the Galaxy
- Introduction to Analog Electronics, and
- Introduction to Multimedia and Web Technology.

In each of these books we introduced 6 annotations - 2 ticks, 2 crosses and 2 circles - on each page, of varying sizes. Results obtained are highlighted in Table 1.

	Before preprocessing		After preprocessing	
	OCR Result	After post processing	OCR Result	After post processing
Errors due to annotations	227	56	50	11
Other errors	71	19	19	06
Total errors	298	75	69	17

Table 1: Results in terms of total number of errors before and after pre and post processing

5. CONCLUSION

In this project, we developed an end-to-end approach to extract textual data from annotated document images accurately. In order to achieve good results, we preprocessed the document images by identifying the affected regions in the document image and then performed inpainting to reconstruct the affected words. The mask of annotation was created using a

constrained undirected path opening. For inpainting, we opted for an image inpainting technique based on the Fast Marching Method. This preprocessing offered the OCR a better document image to work upon. After that the image is passed through the OCR and textual data is extracted. But due to the removal of annotations, some parts of the text are also removed which results in spelling errors in the extracted textual data. To correct these errors we used a seq2seq based spelling correction model. With our proposed method we are able to achieve an accuracy of 94.32%.

As we find from our experiments, our method generalizes well on a plethora of documents with different types of noise it can be safely said that our end to end method for correcting annotations in documents is not just robust but is also versatile.

6. REFERENCES

[1] Rovina and Seema "Detection and removal of hand-drawn annotation lines in scanned image", International Journal of Advanced Computational Engineering and Networking, ISSN: 2320-2106, Volume-4, Issue-5, May.-2016

[2] S. Pratihar, P. Bhowmick, S. Suralt and J. Mukhopadhyay, "Detection and removal of hand-drawn underlines in a document image using approximate digital straightness", Dar '12 Methodology acm, pp. 124-130, 2012.

[3] C.Adak and B. B. Chaudhuri, "An approach of strike-through text identification from handwritten documents", IEEE International conference on frontiers in handwriting recognition, pp.643-648,2014.

[4] Z. L.Bai and Q.Huo, "Underline Detection and Removal in a Document Image using Multiple Strategies", IEEE Pattern Recognition, ICPR Proceedings of the 17th International Conference, vol. 2, pp. 578-581, 2004

[5] S. Das and P. Banerjee, “Gabor filter based hand-drawn underline removal in printed documents”, IEEE Automation, Control, Energy and Systems (ACES), First International Conference, pp. 1-4, 2014.

[6] M. Oba, Y. Nozaki, T. Matsumoto and T. Onoyama, “Underline Removal Method by Utilizing Characteristics of Japanese Business Documents”, TENCON IEEE Region 10 Conference, pp. 1-6, 2009.

[7] J. R. C. Pinto, P. Pina, L. Bandeira, L. Pimentel and M. Ramalho, “Underline Removal on Old Documents”, Springer-Verlag Berlin Heidelberg, vol. 6, pp. 226-233, 2004

[8] T. Saba, A. Rehman,A. Altameem and M. Uddin, “Annotated comparisons of proposed preprocessing techniques for script recognition”, Springer verlag London, pp. 1337-1347, 2014.

[9] T. Singh, M. Mahajan(2014), “Performance comparison of fuzzy c means with respect to other clustering algorithm” International Journal of Advanced Research in Computer Science and Software Engineering, vol.-4, pp. 89-93

[10] Evans D., Zhai C., Tong X., Milic-Frayling, N., OCR Correction and Query Expansion for Retrieval on OCR Data, Clarit trec-5 confusion track report, volume 52, (1996).

[11] Atwell E., Elliittm S., Dealing with ill-formed English Text, The Computational Analysis of English: A Corpus-Based Approach,New York,Longman, (1987).

[12] MaysE., Damerau F.J., MercerR.L, Context-BasedSpellingCorrection, Information Processing and Management, 27, 5, 517-522,(1991).

[13] GoldingR.A.,A Bayesian Hybrid Method for Context-Sensitive Spelling Correction, Proceedings of the Third Workshop on Very Large Corpora, Cambridge, MA. 39-53,(1995).

[14] Golding R.A., and SchabesY.,Combining Trigram-Based andFeature-Based Methods for Context-Sensitive Spelling, pages 71–78, 34th Annual Meeting of the Association for Computational Linguistics, (1996)

[15] Hisao Niwa, Kazuhiro Kayashima, Yasuham Shimeki,Post Processing for Character Recognition Using Keyword Information, IAPR Workshop on Machine Vision Applications,Dec. 7-9, Tokyo, (1992).

[16] Lon-Mu Liu, Yair M. Babad, Wei Sun, Ki-KanChan, Adaptive Post-Processing of OCR Text via Knowledge Acquisition, Proceedings of the 19th annual conference on Computer Science, (1991).

[17] Kazem Taghva, Julie Borsack, Allen Condit,Results of Applying Probabilistic IR to OCR Text, proceedings of the seventeenth annual international ACMSIGIR conference on Research and development in information retrieval, (1994).

[18] Michael L. Wick, Michael G.Ross, Erik G. Learned-Miller, Context-Sensitive Error Correction: Using Topic Models to Improve OCR,In the proceedings of the 9th International Conference on Document Extraction and Analysis,page(s): 1168 -1172,ISSN: 1520-5363, (2007).

[19] K. Kise, T. Shiraishi, S. Takamatsu, K. Fukunaga, A method of Post-Processing for Character Recognition based on Syntactic and Semantic Analysis of Sentences, Journal of Systems and computers,Japan,ISSN 0882-1666, CODEN SCJAEP, vol. 27, no9, pp. 94-107 (1996).

[20] J. Hull,Incorporating Language Syntax in Visual Text Recognition with a Statistical Model,IEEE Transactions on Pattern Analysis and Machine Intelligence, 18(12), (1996).

[21] Ilya Sutskever, Oriol Vinyals, and Quoc VV Le. Sequence to sequence learning with neural networks. In Advances in Neural Information Processing Systems, pages 3104–3112, 2014.

[22] M. Schuster and K. K. Paliwal, "Bidirectional recurrent neural networks," in IEEE Transactions on Signal Processing, vol. 45, no. 11, pp. 2673-2681, Nov. 1997, doi: 10.1109/78.650093.

[23] Telea, Alexandru, 2004/01/01, An Image Inpainting Technique Based on the Fast Marching Method, 10.1080/10867651.2004.10487596, Journal of Graphics Tools

[24] Ashish Vaswani, Noam Shazeer, Niki Parmar, JakobUszkoreit, Llion Jones, Aidan N Gomez, LukaszKaiser, and Illia Polosukhin. 2017. Attention is all you need. In Advances in Neural Information Processing Systems, pages 6000–6010.