Towards Scalable Applications for Handwritten Documents

Thesis submitted in partial fulfillment
of the requirements for the degree of

Master of Science in ***Computer Science and Engineering*** *by Research*

by

Vijay Bapanaiah Rowtula
201350873
`vijay.rowtula@research.iiit.ac.in`

International Institute of Information Technology, Hyderabad
(Deemed to be University)
Hyderabad - 500 032, INDIA
June 2019

International Institute of Information Technology
Hyderabad, India

CERTIFICATE

It is certified that the work contained in this thesis, titled "Towards Scalable Applications for Handwritten Documents" by Vijay Bapanaiah Rowtula, has been carried out under my supervision and is not submitted elsewhere for a degree.

Date

Adviser: Prof. C.V. Jawahar

To

My Family

for their unconditional love and support

Acknowledgments

As I submit my thesis, I would like to thank all the people who have supported me and motivated me throughout my journey at IIIT.

Firstly, I would like to express my sincere gratitude to my advisor Prof. C. V. Jawahar for the continuous support of my MS study and related research, for his patience, motivation, and immense knowledge. His guidance helped me learn the importance of dedication, consistency, and patience required in the field of research. I could not have imagined having a better advisor and mentor for my first steps in research. My sincere thanks also goes to Dr. Anoop M. Namboodiri, Dr. Naresh Manwani, and Dr. S. Bapi Raju, who provided me an opportunity to work on various projects, and who gave access to the research facilities. I thank my mentor, Praveen Krishnan for the stimulating discussions, for the sleepless nights we were working together before deadlines, and for all the support and guidance provided in the last three years.

I would also like to thank my colleagues Ajoy Mondal, Minesh Mathew and Viral Parekh. Without their precious support it would not be possible to conduct this research. It was a good learning opportunity to work as a sysadmin for Shiksha AMS portal with Varun, Mohan and Ved Prakash. Working with Varun on a couple of project enhancements was a great experience. I thank him for his help and suggestions in my project work. Special thanks to Siva, Silar, and Rajan for all of their help on various occasions. I profusely thank my friends Vijay Saradhi Indurthi and Subba Reddy Oota who made my journey at IIIT memorable.

Last but not the least, I express my deepest gratitude toward my parents, R. Balagangadhara Tilak and R. Raja Kumari, my wife Madhuri and my son Tanush, for their unconditional support in all my endeavors.

Abstract

Even in today's world, a large number of documents are generated as handwritten documents. This is specially true when the knowledge/expertise is captured conveniently with availability of electronic gadgets. Information extraction from handwritten document images has numerous applications, especially in digitization of archived handwritten documents, assessing patient medical records and automated evaluation of student handwritten assessments, to mention a few. Document categorization and targeted information extraction from various such sources can help in designing better search and retrieval systems for handwritten document images. Information extraction from handwritten medical records written in ambulance for doctor's interpretation in hospital, reading postal address to automate the letter sorting are examples where document image work flow helped in scaling the system with minimal human intervention. In such work flow systems, images flow across subjects who can be in different locations. Our work is motivated with the success of these document image work-flow systems that were put into practice when the handwriting recognition accuracy was unacceptably low. Our goal is to bring scalability in handwritten document processing which can enhance the throughput of the analysis by employing multitude of developments in document image space.

In this thesis, we initially focus on presenting a document image workflow system that helps in scaling the handwritten student assessments in a typical university setting. We observed that this improves the efficiency since the book keeping time as well as physical paper movement is minimized. An electronic workflow can make the anonymization easy, alleviating the fear of biases in many cases. Also, parallel and distributed assessment by multiple instructors is straightforward in an electronic workflow system. At the heart of our solution, we have (i) a distributed image capture module with a mobile phone (ii) image processing algorithms that improve the quality and readability (iii) image annotation module that process the evaluations/feedbacks as a separate layer.

Further, we extend our work by proposing an approach to detect POS and Named Entity tags directly from offline handwritten document images without explicit character/word recognition. We observed that POS tagging on handwritten text sequences increases the predictability of named entities and also brings a linguistic aspect to handwritten document analysis. As a pre-processing step, the document image is binarized and segmented into word images. The proposed approach comprising of a CNN-LSTM model, trained on word image sequences produces encouraging results on challenging IAM dataset.

Finally, we describe an effective method for automatically evaluating the short descriptive handwritten answers from the digitized images. Automated evaluation of handwritten answers has been a challenging problem for scaling education system for many years. Speeding up the evaluation still remains as the major bottleneck for enhancing the throughput. Our goal is to assign an evaluation score that is comparable to the human assigned scores. Our solution is based on the observation that a human evaluator judges the relevance of the answer using a set of keywords and their semantics. Since reliable handwriting recognizer are not yet available, we attempt this problem in the image space. We model this problem as a self supervised, feature based classification problem, which can fine tune itself for each question without any explicit supervision. We conduct experiments on three different datasets obtained from students. Experiments show that our method performs comparable to that of human evaluators.

With these works, we attempted to bring state-of-the-art enhancements in handwritten document analysis and deep learning into scalable applications which can be helpful in the field of education.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

Document images are images with rich textual content along with other important content like images or equations. Even though most aspects of today's world are dominated by usage of software and computers, a large number of documents are still generated as handwritten documents. In many ways, handwritten documents are still less restrictive than its digital counterpart and has many advantages both functionally and creatively. Students home works and assessments capture the work and learning abilities where his/her thoughts are freely expressed without the constraints seen in electronic sources like tablets or mobiles. Similar to the text documents, digitization of handwritten documents can help in better analysis without the hindrances on location or expertise and can also enable us with better organization and archival of handwritten document images. With the availability of cheaper electronic gadgets, digitization of knowledge is now even more convenient.

Document images also require good document workflow system for obtaining better throughput especially when document images are handwritten. Current document repositories provide solutions as organized storage but do not provide systems or plugin to integrate research advances in document image analysis. This is especially lacking in today's document content repositories which have content extraction plugins for text file formats like PDF and word docs but not for handwritten document images. Information extraction from handwritten medical records [1] written in ambulance for doctor's interpretation in hospital, reading postal address [2] to automate the letter sorting are examples where document image work flow helped in scaling the system with minimal human intervention. In such work flow systems, images flow across subjects who can be in different locations but still can contribute to its processing. A postal automation module in USA can take help of a person in Asia to recognize the address block and still continue to be efficient. Often such image work flow systems become intelligent over time and need minimum human help. Our work is motivated with the success of these document image workflow systems that were put into practice when the handwriting recognition accuracy was unacceptably low. We initially present a scalable workflow system for handwritten document images which can plug-in latest research in the field of handwritten document image analysis. Workflow system with plug-in enhancements can help the framework to be updated with research work across the document analysis domain.

We attempted to develop few plug-in applications which can be helpful in the field of education. Some of the plug-in enhancements are described in chapter 2, which can enhance the experience of handwritten document visualization and archival for educational institutions. Next, we focused specifically on one such key aspect, to detect keywords directly from handwritten document images, without transcribing the document to text. This is different from word spotting where desired keyword is spotted across set of documents, where as keyword detection finds all the relevant keywords in a given handwritten document image. We attempted to tag parts-of-speech, noun phrases and named entities on handwritten document images.

We next describe about another application which is useful in current educational system. Automated evaluation of student assessments has been a challenging problem for scaling education system for many years. With the availability of digital content and cheaper personal computers, some educational institutions are opting for online text assessments, multiple choice questionnaire or optical markers. In recent years, the number of online educational applications has been growing steadily - including intelligent tutoring systems, e-learning environments, distance education and massive open online courses (MOOCs). But most educational institutions still prefer the traditional system of handwriting based assessments across all levels of students education, as it rightly captures the students actual work and thought process. Students natural language input can be differentiated, particularly in content and writing style and their overall grasp on concept. Teachers spend lot of time providing precise feedback on assignments and grading student responses to assessment questions. Though numerous research works were published with focus on automating text based assessments (ASAG) or short essay answers (AES), there were very few attempts focused on handwritten short answer assessments. We designed a solution that helps in automatic evaluation of the answers. The automated evaluation method is enhanced using features such as semantic query expansion and keyword or named entities spotting directly on handwritten document images. Our solution can be integrated across all levels of education, which can cut down the assessment time of teachers and help them utilize same time in other educational activities.

1.1 Problems of Interest

There are vast number of open problems associated with each of the domains associated with handwritten document image analysis. Here in this thesis, we have selected a few challenges and tried to solved the problem and proposed some real world applications emerged from the field of Document image analysis. We have restricted our work in the area of designing scalable solution to document work flow system and also provided solutions to tougher problems like NLP on document images and automating the evaluation processing of students assessments. However the applications are open to further enhancements which could be useful to solve problems in other domains as well. In the following sections we have provided detailed overview about the problems of our interest.

1.2 Scalability in Education

Regular, personal feedbacks are critical to learning. Traditionally, this has been achieved through qualitative/quantitative assessments and through home works. Over time, electronically created and formatted documents have crept into the system which limited the effectiveness of assessment. Managing student assessments consume a significant portion of the effort of a teacher. With the need to scale, modern assessment systems are slowly moving towards solutions that can automate the evaluation process. Examples include multiple choice questions, fill in the blanks, matching two sets and output based computer program evaluation. Personal touch of the assessment process is also disappearing with the penetration of Internet and electronic solutions. We now have a contradicting requirement of scalability and effectiveness. We make a contribution in assessment space with a document image workflow system that can bring the advantages of the electronic workflow into the world of physical paper. We present a system that supports several assessment formats with special emphasis on handwritten assessments. The system also provides plug-in support for enhancements to integrate or update further innovations in student assessment space.

The focus here is to demonstrate a scalable "paperless grading" system for handwritten assessments which allows electronic submission and on-screen grading of the assessments with high transparency between instructors and students. We briefly discuss the plug-ins added to enhance the paperless grading experience. With this system, we expect to increase the through-put for the instructors and their time can be utilized in other productive activities.

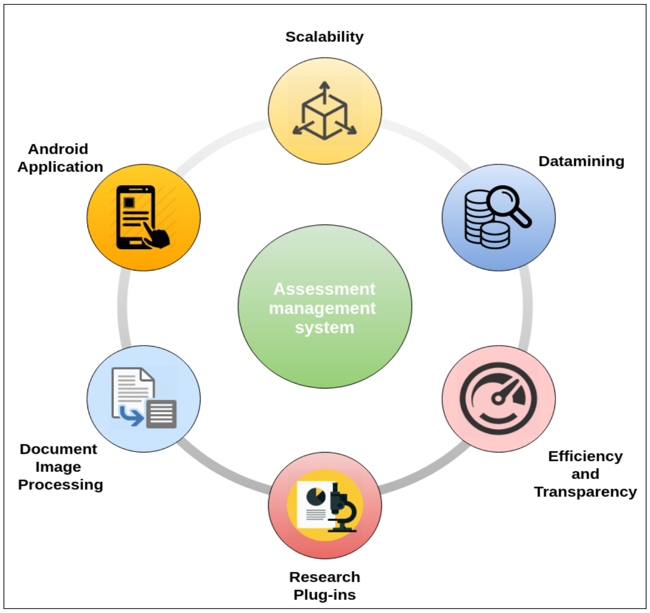
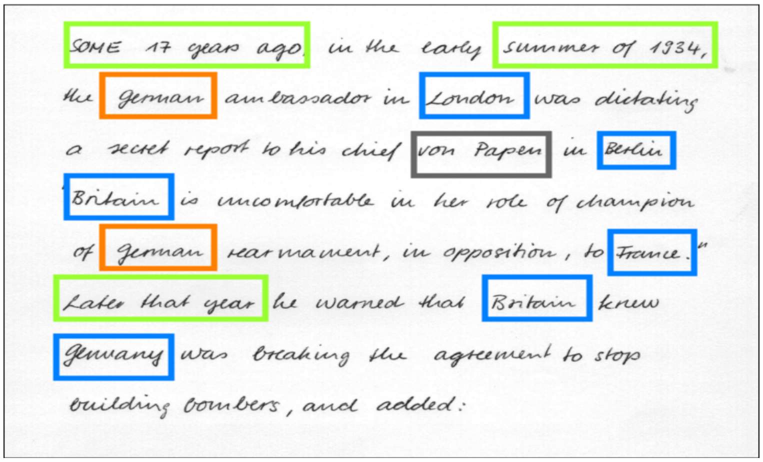

Scalability
Android Application
Datamining
Assessment management system
Document Image Processing
Efficiency and Transparency
Research Plug-ins

Figure 1.1 The image highlights some of the key aspects we focused on, to develop our document workflow system. These are the differentiating factors which sets apart our system from existing learning management systems.

1.3 NLP on Handwritten Document Images

Semantic annotation of handwritten documents, especially spotting keywords using POS tags or NER is relatively a newer problem with very few works emerging on this front. A traditional approach to information extraction would be to first transcribe the text, and then use dictionaries, grammars or some other NLP (Natural Language Processing) techniques to detect named entities. Without character/ word recognition, POS tagging and named entity recognition from a document image is quite difficult because NLP-based knowledge can hardly be used in such a situation. However, such detection is essential where linguistic knowledge of text cannot be used due to the poor performance of handwritten text recognition engines. Named entity recognition is an information extraction problem consisting in detecting and classifying the keywords into pre-defined categories such as the names of people, streets, organizations, dates, etc. It can also be seen as the semantic annotation of text elements.

We attempt to fill the gap by proposing an approach for POS tagging and NER without handwriting transcription. Recent work in deep neural nets suggests combining multiple models trained on the same dataset for subtasks individually, into a single model for a similar or better performance due to lower error propagation from the different stages. The contribution of this work is to show generalization with a similar or improved performance of a unified end-to-end model without separating the sequence of sub-processes involved, thereby avoiding error propagation. Identifying named entities using noun phrases from POS tags can also be greatly helpful for keyword-based document retrieval. Detecting named entities irrespective of its structural and positional characteristics (e.g. uppercase or lowercase letters) is an advantage of our approach. As a pre-processing step, we choose a handwritten dataset with segmented words and POS tag annotations. It helped us focus only on the aspect of POS and named entity tagging on handwritten word images rather than the problem of word segmentation from handwritten documents.



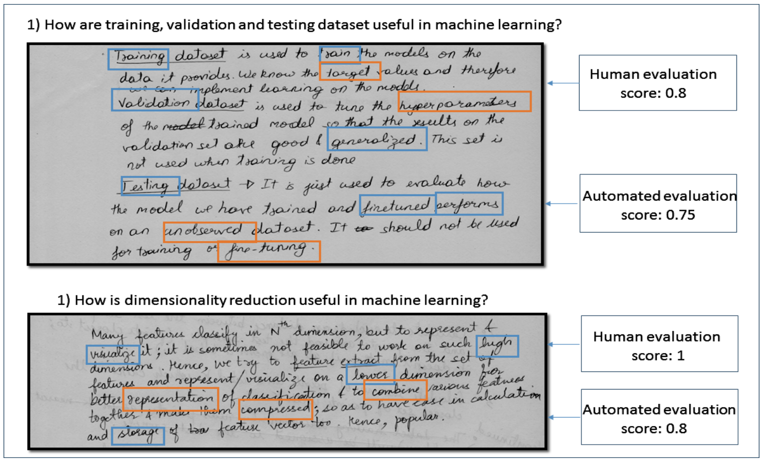
SOME 17 years ago in the early summer of 1934,
the German ambassador in London was dictating
a secret report to his chief von Papen in Berlin
"Britain is uncomfortable in her role of champion
of German rearmament, in opposition to France."
Later that year he warned that Britain knew
Germany was breaking the agreement to stop
building bombers, and added:

Figure 1.2 We are interested in spotting the named entities in Handwritten Document Images. The figure depicts an example where keywords are classified into various named entities as highlighted by their colors - locations are in blue, date in green, persons in grey and nationalities in orange.

1.4 Automated Evaluation

Automated evaluation of answers is an active area of research in the text domain. While computer based testing is becoming the standard for entrance and online assessments, handwritten responses are still the principal means in schools and college examinations. Assessing large numbers of handwritten papers is a relatively time consuming and monotonous task. There is a need to speed up and enhance the process of grading handwritten responses, while maintaining simplicity and cost effectiveness of system. A multitude of measures for computing similarity between the true answer and the candidate answer have been proposed in the past based on surface level and semantic content features [3, 4]. Various linguistic aspects of the sentences were covered using WordNet [5], corpus-based features [6], Word2Vec [7], alignment-based features [8] and literal-based features [3]. Commercial systems (E.g. [9]) use a combination of statistical and natural language processing (NLP) techniques to extract linguistic features and use them in comparing answers. Though the text based automatic evaluation is nearing the reliable deployment in the university education system, handwritten answers are not yet amenable for their processing.

Evaluation of handwritten answers needs significant advance in computer vision algorithms (e.g. text segmentation and recognition). A natural direction to evaluate the handwritten answers is to recognize the textual content and then exploit the advances in the text based automatic evaluation. While the printed text can be reliably recognized with optical character recognizer (OCR), offline handwritten text recognizer for unconstrained vocabulary are not robust enough for the practical use due to the inherent complexity of a handwritten word image. We present a word spotting based automatic evaluation solution based on the deep learned features.


1) How are training, validation and testing dataset useful in machine learning?
Human evaluation score: 0.8
Automated evaluation score: 0.75
1) How is dimensionality reduction useful in machine learning?
Human evaluation score: 1
Automated evaluation score: 0.8

Training dataset is used to train the models on the
data it provides. We know the target values and therefore
~~we can~~ implement learning on the models.
Validation dataset is used to tune the hyperparameters
of the ~~model~~ trained model so that the results on the
validation set are good & generalized. This set is
not used when training is done
Testing dataset → It is just used to evaluate how
the model we have trained and finetuned performs
on an unobserved dataset. It ~~co~~ should not be used
for training or fine-tuning.

Many features classify in N^th dimension, but to represent &
visualize it; it is sometimes not feasible to work on such high
dimensions. Hence, we try to feature extract from the set of
features and represent/visualize on a lower dimension for
better representation of classification & to combine various features
together & make them compressed; so as to have ease in calculation
and storage of ~~too~~ feature vector too. Hence, popular.

Figure 1.3 We are interested in assigning a quantitative score to a handwritten answer that match with the score assigned by a human evaluator. Figure depicts two examples where answers from datasets evaluated by human evaluator and by our assistive evaluation framework. To automatically evaluate, we match keywords that are present in the textual sample answer (blue box) as well as those that are not directly provided (orange).

1.5 Major Contributions

- **Problem:** Scaling workflow system for handwritten student assessments.
  For this problem we designed a platform for upload and assessment of students handwritten answers with plug-in architecture for adding up-to-date research enhancements is document analysis space.

- **Problem:** POS Tagging and Named Entity Recognition on Handwritten Documents.
  We have proposed an approach comprising of a CNN-LSTM model, to perform POS and Named Entity tagging directly on word image sequences with out transcribing them to text.

- **Problem:** Automated Evaluation of Handwritten Assessments.
  This work investigates the ability to model the problem as a self supervised, feature based classification problem, which can fine tune itself for each question without any explicit supervision.

1.6 Thesis Outline

The chapters in this thesis are organized as follows. First we have discussed our contribution in the domain of scalability in handwritten document image workflow platform and its applications in education system Then we have discussed the natural language processing on handwritten word images and its application in automating the evaluation of handwritten assessments

Chapter 2: Scaling Handwritten Student Assessments with a Document Image Workflow System

Chapter 3: POS Tagging and Named Entity Recognition on Handwritten Document

Chapter 4: Towards Automated Evaluation of Handwritten Assessments

Chapter 2

Scaling Handwritten Student Assessments with a Document Image Workflow System

Student-teacher interactions, mentoring and feedback are vital to the process of learning. This is achieved through classroom teaching, communication and through various kinds of assessments. Our teaching system traditionally has a strong inclination towards handwritten assessments, which reflect students thought process and creativity. Hence, handwritten document and handwriting in general, forms integral part of our education system. We are accustomed to manual assessments where instructors manually evaluate both students homeworks and assessments. This effects the productivity of both teacher and student since assessments consume a substantial part of the effort of a teacher and students have to wait for feedback. With the need to scale, modern assessment systems are slowly moving towards solutions that can automate the evaluation process. Examples include multiple choice questions, fill in the blanks, matching two sets and output based computer program evaluation. Over time, electronically created and formatted documents have crept into the system which limited the effectiveness of assessment. With the penetration of the Internet and electronic solutions, personal touch of the assessment process is also disappearing.

We make a contribution in assessment space with a document image workflow system that can bring the advantages of the electronic workflow into the world of physical paper. We hope that this will also enable research that can scale the handwritten assessments by processing document images in the near future. With the advent of Web 2.0 and MOOCs, e-learning platforms have gained popularity and have made a profound impact in the field of education. Initially, schools and universities embraced the convenience of paper-less computer-based teaching and assessments where the whole system of creating tutorials, video classes, assessments and its reminders, calendar events were moved to computer organized system from manual work. Current virtual learning environments, also called as learning management systems (LMS) typically provide tools for assessment, communication, uploading of content, administration of students, questionnaires, tracking tools, wikis, blogs, chats, forums, etc. over the Internet. Learning management system (LMS) is the current approach to e-learning. Learning in LMS is organized as courses, and it usually serves as the online platform for course syllabus releasing, handouts distribution, assignments management, and course discussion to students, teachers, TAs who are the members of the same course[10]. LMS such as Blackboard, Moodle [11], and Sakai has been used by numerous universities all over the world to support and improve learning of their students; it is primarily designed for course management purpose and has constraints in areas of assessment management and scalability [12]. However, they have few drawbacks. The primary limitations of LMS include limited interaction channel and collaboration manner between learners and educators [13][14], restricted interaction and collaboration scope within courses and limited support for handwritten assessments. Since schools and universities still follow traditional modes of teaching and assessments where students depend extensively on paperwork for home works and assessments, some support for handwritten documents can help current LMS system to better penetrate the present teaching system. Handwritten assessments have been a dominant format to create and evaluate students. It shows the organization of thoughts, original expressions and creativity in comparison to the electronically formatted solutions that does not show the fingerprints of a student. The time spent by a student in writing the home works and assessments tend to translate into long-term memories, helping students in better retention of the subject. It is observed that for handwritten assessments, students do not receive any detailed feedback quickly for it to be helpful enough in their next assessment, because of the time delay involved in distribution, evaluation, entry of grades etc.

In this chapter, we present a system that supports several assessment formats with particular emphasis on handwritten assessments. The system also provides plug-in support for enhancements to integrate or update further innovations in student assessment space. A conceptual explanation of the system is shown in Figure 3.1. We also describe the integration of our application with a mobile app designed extensively to increase the through-put of the student while uploading the handwritten assessments. Students digitize the handwritten document with a mobile phone-based interface. The app has minimum operations to select, crop, rotate and upload the document the student intends to upload to a central server. Instructors can grade/assess by annotating the images online using a web-based interface. This simple yet effective connect between the physical paper world, and electronic workflow makes our solution effective and efficient.

2.1 Document Image Work Flow Systems

The history of the application of computers to education is filled with broadly descriptive terms such as computer-managed instruction (CMI), and integrated learning systems (ILS), computer-based instruction (CBI), computer-assisted instruction (CAI), and computer-assisted learning (CAL). These terms describe drill-and-practice programs, more sophisticated tutorials, and more individualized instruction, respectively.

The first fully featured Learning Management System (LMS) was called EKKO, developed and released by Norway's NKI Distance Education Network in 1991. The current top three LMS by number of installations were Blackboard, Moodle and Canvas. These LMS mostly cater to the needs of end-to end learning curriculum which includes student management, courses management and students assessment

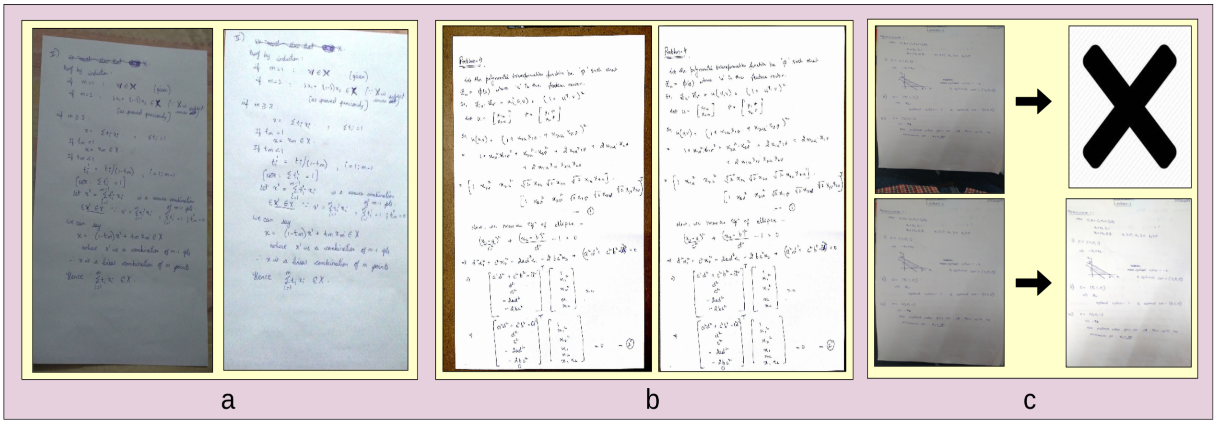


Figure 2.1 Example of original and processed handwritten assessments before being sent for evaluation. Sets (a), (b) contain pre and post processed handwritten assessments. Set (c) shows assessment rejection due to inconsistencies (top) and a better image was uploaded by student and was processed (bottom). We can notice the border, color and brightness rectification in all three image sets.

which include quizzes and code based evaluations. Several assessment management systems has been developed particularly for student assessments. There are products like OpenEduCat, Skolaro etc which focus on student assessments. Blackboard Learn is a virtual learning environment and course management system which features course management, customizable open architecture, and scalable design that allows integration with student information systems and authentication protocols. OpeneduCat is a comprehensive open source ERP for educational institutes with an easy to use student information management system, faculty management, course management, a helpful enrollment and examination management along with integrated financial management. Skolaro is cloud based, integrated knowledge platform with collaboration, data analytics and machine learning at core of offering. Moodle is a learning platform designed to provide educators, administrators and learners with a single robust, secure and integrated system to create personalized learning environments. Though they succeeded in automating online assignments, automating uploads of traditional handwritten texts and its evaluation at large scale were two major issues highly ignored.

Information extraction from handwritten medical records [1] written in ambulance for doctor's interpretation in hospital, reading postal address [2] to automate the letter sorting are examples where document image work flow helped in scaling the system with minimal human intervention. In such work flow systems, images flow across subjects who can be in different locations. A postal automation module in USA can take help of a person in Asia to recognize the address block and still continue to be efficient. Often such image work flow systems become intelligent over time and need minimum human help. Our work is motivated with the success of these document image workflow systems that were put into practice when the handwriting recognition accuracy was unacceptably low.

The focus of this chapter is to demonstrate a scalable paperless grading system for handwritten assessments which allows electronic submission and on-screen grading of the assessments with high transparency between instructors and students. In Section 2.2, we introduce our document workflow system, its image processing modules and provide a brief overview of system architecture. In Section 2.3, we describe our experience using the workflow system. We also explain how the recent advances in handwritten document analysis will be integrated into our workflow system, opening up new avenues in research which can impact education.

2.2 Assessment Management System

We now start by looking at what can happen in a typical classroom scenario. Faculty member provide questions and students bring their solutions to classroom or submit them at a fixed location. A teaching assistant assigned by the faculty member or the faculty member herself (instructors) grade the assessments and provide quantitative and/or qualitative feedback. Finally the grades are available to students, after a brief discussion between students and instructors about evaluation corrections. In the following sections, we explain how our solution was designed to troubleshoot the pain points faced by instructors and students during the workflow process.

2.2.1 Design Goals

We started with the following set of goals:

- Make the overall student assessment process efficient by removing paper movement, paper arrangements (e.g. sorting pile of papers by student IDs) and additional data entry (manual entry of scores into a database explicitly). This can greatly reduce manual document management traditionally followed in schools and colleges. The cumbersome movement from students assessments on papers to the collection, redistribution and entry of evaluation score in greatly reduced in our system and hence can increase the through-put of both students and instructors.

- Bring correction/evaluation electronically as an extra annotation layer. This should enable parallel, distributed and multiple grading of the same student assessment. This is implemented by providing an extra layer of user interface where the instructors can conveniently evaluate the students handwritten answers directly on computer. Students and instructors can start the discussion immediately. This promotes transparency among students and instructors.

- Incorporate a set of computer vision methods required to meet the immediate goal and keep the design open to introduce advanced image recognition modules at a later stage. This is the primary advantage of our platform where update research in computer vision directly translates to the application hence acting as a large scale testing platform for research output. This also helps in automating some important tasks like writer identification, plagiarism detection which can be helpful in university setting.

- A system that can learn, improve and adapt over time. For example, common errors/feedbacks are mined from the annotations and displayed on novel situations, thus minimizing the effort. The feedback acts as a control loop where the back-end algorithms considers them as additional training data and improves the performance over time, thus decreasing such errors at a later point of time.

2.2.2 Document Image Processing

In our assessment evaluation process, student first uploads camera-captured document images using an android application (discussed in Section 2.2.4). It is a known fact that camera-captured images are prone to various degradation such as inadequate lighting, shadows, blur and camera flash at times. Such degradation often lead to difficulties in analysis at subsequent stages of image processing. For example, degradations may result in a significant drop in the performance of Optical Handwriting Recognition (OHR), word spotting and other handwritten document analysis tasks, resulting in unrecoverable information loss.

The degradation introduced can be classified into *(i) Character level - with broken characters, touching, skewed or curved handwriting, (ii) Page level - margin noise, salt-and-pepper, ruled line, warping, curling, skew, blur or translation*. We focused on rectifying page level degradation.

2.2.2.1 Capturing Handwritten Assessments

Though the students in traditional learning management systems have the comfort of submitting the handwritten assessments from any location, the assessments still have to be compressed (zipped) and uploaded to a server. Instructors will have to download the file and then evaluate the handwritten or other file based assessments. For handwritten assessments, our workflow solution includes an android application which is used by students to take pictures of the assessments and upload them to server immediately. This can be very helpful in scenarios such as a surprise or spot assessment in class room. The android application tries to qualify the images based on the visual aesthetics of the uploaded handwritten document image. We used methods described in [15, 16], which uses a set of local character level features and global page level features to arrive at a quality score. The android application will reject the images with lower than a permissible score on distortions as seen in Figure 3.1. In such cases, student has to re-upload a proper image of his handwritten assessment. Legible images are finally uploaded to server with the consent of student.

2.2.2.2 Dewraping Camera-Captured Images

Compared to scanners, mobile cameras offer convenient, flexible, portable, and non-contact image capture, which enable better throughput in a document workflow management system. However, camera-captured documents may also suffer from distortions caused by non-planar document shape and perspective projection, which can lead to failure of current OCR/OHR technologies. The images were rectified based on the method explained in [17]. These methods share a similar hierarchical problem decomposition: (i) Split the text into lines. (ii) Find a warp or coordinate transformation that makes the lines parallel and horizontal. Though the cited methods were modeled for printed text, we observed that same methods worked well for camera-captured handwritten document images. A sample of dewrapped images can be seen in Figure 2.2.

2.2.2.3 Rule Line Removal from Handwritten Assessments

Some of the students submit their assessments in rule lined pages, as shown in Figure 2.2. Rule lines - both horizontal and vertical, should be removed to ensure better analysis at subsequent stages of image processing. We adapted methods described in [18] which uses rule line detection using Horizontal Projection Profile (HPP) and Hough Lines (HL). The steps involved are: (i) De-skew the image using method described in earlier section (ii) Extract the location of horizontal lines using combination of HPP and HL (iii) Remove the lines from the de-skewed version of original document image and (iv) Reconstitute the missing pixels. Image (b) in Figure 2.2 shows original camera-captured document image and its rectified version.

2.2.2.4 Annotation of Images

Our solution allows on-screen evaluation of uploaded handwritten assessments. The instructor can highlight, annotate and comment on document images. These annotations are saved separately along with its image coordinate details. Since these annotations are immediately available to the students, they can immediately start a discussion with the instructors. The keywords from questions, assessment image and discussions together form a rich set of evaluation annotations for an assessment platform, which can be mined for patterns and reused while evaluating a similar assessment of other students.

Though these are experimental features, they demonstrate the extensibility of our document workflow platform in handwritten assessment space.

2.2.3 Other Features

2.2.3.1 Easing Assessments

Our system design is focused on the task of decreasing the execution time of student assessments. From the creation of questions to final grading by instructors, our workflow system simplifies the complete process, by moving most of the manual procedures to web application. Students can either upload the handwritten answers using a mobile android application (Figure 2.3) or upload an answer file using web interface or even directly type in the answer. For code evaluations, students can upload the code to the portal and evaluation is completed online, as explained in Section 2.2.4. Text and image based answers are evaluated on-screen using our portal.

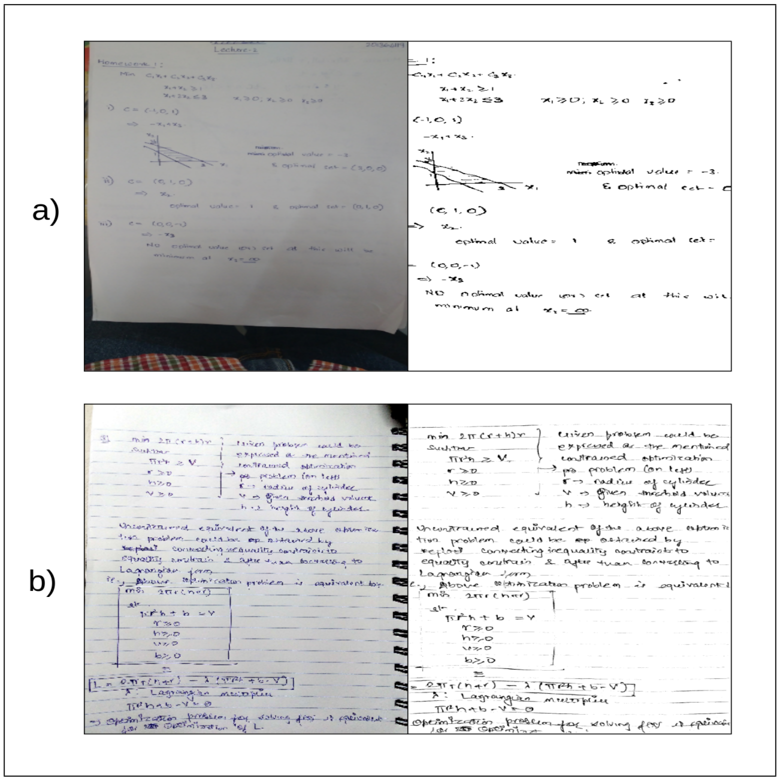


Figure 2.2 Sample Document Images rectified using Image processing. First row (a) has original image and de-warped document image free from distortions (shadows and bends). The second row (b) has original image with rule lines / bad illumination and de-warped document image free from distortions.

2.2.3.2 Data Mining in E-learning

The application of data mining in e-learning systems is an iterative cycle. The mined knowledge should enter the loop of the system and enhance learning as a whole, and facilitate filtering of mined knowledge for decision making. Our solution uses simple data analysis to observe student's behavior and assist instructors in detecting possible shortcomings to incorporate improvements. It mines the data and creates report on student assessment submission delays, highly performing and under performing teaching assistants, forums harboring negative discussions and other similar vital stats. A weekly status update by email is sent to both students and instructors with consolidated stats.

Thus, the system helps in identifying the achievement gaps among students and tutors alike, measures the effectiveness of a course, academic program or learning experience over the course duration.

2.2.3.3 System Transparency

This is implemented by processes such as double blind assessments, peer review of evaluations, discussion forums, dashboards by profile hierarchies and weekly status updates by email. The double blind procedure makes sure of unbiased evaluations and discussion between students and instructors. The

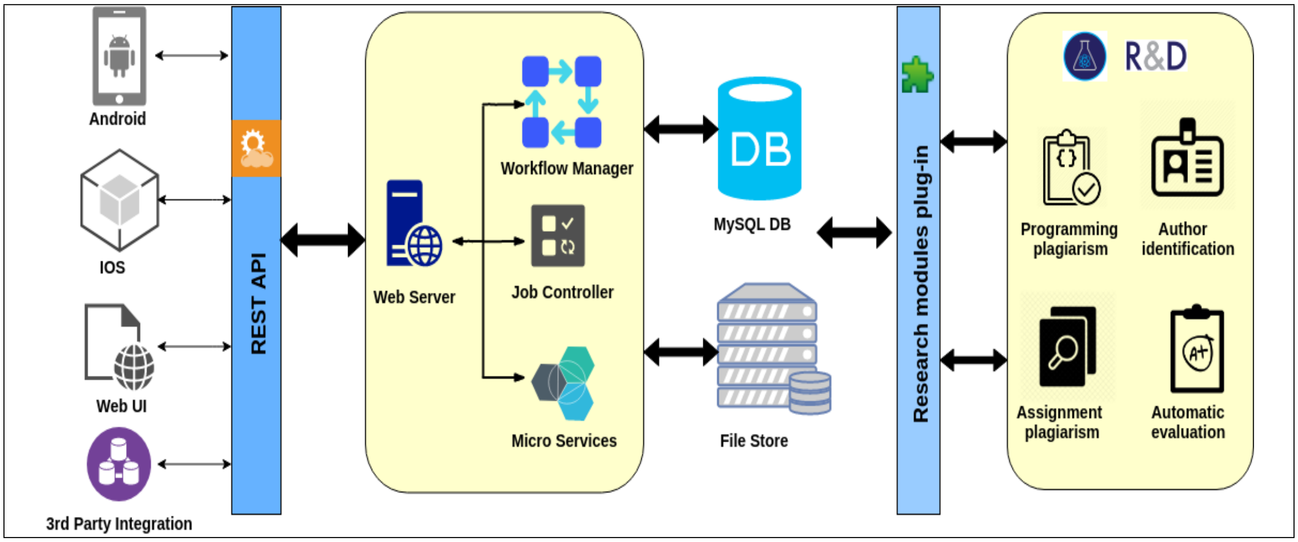


Figure 2.3 System architecture and workflow of Assessment Management System.

queries and discussions on evaluations can be monitored down the work flow hierarchy. Based on roles, the login page has dashboard which summarizes important updates to students and instructors. The performance of students and TAs are mined from databases which are available on teacher dashboards, hence promoting transparency throughout the workflow.

2.2.4 System Architecture and Implementation

The Assessment Management System architecture was designed with modularity, scalability and extensibility in mind. Figure 2.3 describes the software architecture of the system and shows the modules therein. Some of the key aspects are discussed next.

2.2.4.1 Scalability

The ease of use for assessments, described in Section 2.2.1 brings up a new challenge - scalability. Platform is massively scalable due to use of open source technologies such as Django, MySQL and Docker [19]. It is scalable in terms of hosting number of courses, enrolling and managing large number of students, assessments etc. Currently, more than 15 courses were hosted on our document workflow system, with students count varying from 30 to 150 per course. Even the possible bottlenecks for automated code evaluations are addressed using docker containers. A docker container is a virtual sandbox to create and manage resource per user. Pre-defined resources are allocated per user using docker, hence avoiding system downtime due to possible hacking or resource consumption beyond permissible limits. Another possible bottleneck is handwritten assessment evaluations. This is addressed by on-screen evaluation provided by an intuitive user interface to navigate through assessments.

2.2.4.2 Mobile Application

An android application was designed to work with REST API, which also supports assisted image capture and image corrections. This application supports submission of hand-written answers, by allowing the capture of the hand-written document using the camera of the mobile device. This android application is currently being extended for touch screen devices to speed-up assisted evaluation as explained in Section 2.3.2.

2.2.4.3 Code Evaluation Module

Code evaluation module supports automated evaluation of programming assessments. It supports accepting source code/code snippets as answer submissions and evaluation of those submissions in secure and contained environments. It uses various sandbox and container technologies to run these codes in a safe environment and supports popular programming languages like C, C++, Python, Java, etc. Instructors can customize evaluations by adding custom code snippets during creation of programming questions.

2.2.4.4 Research Plug-in

Various top research papers in handwritten and programming assessment space are evaluated and converted into research modules. These modules are first evaluated on smaller test sets and are finally plugged into the system. We have focused specifically on handwriting and programming space to assist the evaluators dealing with courses containing handwriting and programming assessments. Various in-house research projects are also integrated into the system. The research modules are discussed in detail in Section 2.3.2.

2.2.4.5 Peer Review Module

Our document workflow system can support peer review of answer submissions to enhance or replace evaluation by a dedicated evaluator. The anonymity which this system can provide increases the reliability of the peer-review process as a whole. The time required for distribution and collection of the submissions, which makes up the bulk of the time wasted during a regular peer-review process, is saved by using such a system. This makes peer-review a feasible option even for assessment evaluation.

2.2.4.6 Third Party Integrations

Our document workflow system provides a set of robust REST API (web-services) that provides an easy usability and extensibility of the platform. The APIs can be used to integrate our application to any third party systems and websites. The advantages of such an integration is many-fold. It is possible to use the research modules of our system in 3rd party applications and websites. Any on-line teaching

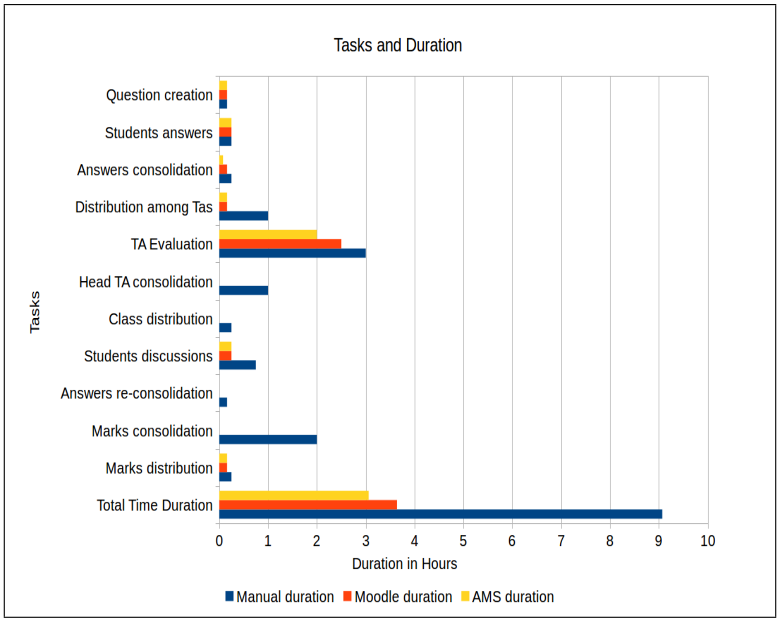


Figure 2.4 Graph shows the effectiveness of our document workflow system compared to manual and Moodle based student evaluations in handwritten assessments space

platform will be able to integrate our document workflow system, as an extension to manage their assessments.

2.3 Experience and Discussions

2.3.1 Experiences

We tested our document workflow system in the real world for 15 university courses. A total of 101 assessments were posted on the platform so far. The assessments contain 607 questions out of which 540 are handwriting based. The total number of student answers is 29300 out of which, the total number of handwritten answers is 20200. We receive feedback from the tutors and students after every course for improvements. The feedback is based on the 6 different aspects of usage of the document workflow system - *interactivity, tutor support, peer support, user-friendly, time management and insights*. Student can also report bugs and enhancement requests. The feedback so far indicated that all students experienced an optimal learning environment and most often suggested improvements in peer-support and interactivity.

2.3.1.1 Class Room Experiment

We have also conducted an experiment to validate the effectiveness of usage of our workflow system for handwritten assessments. As described in Table 2.1, a set of three questions from *Optimization*

Methods course was provided to a class of 127 students with 4 teaching assistants. Students were divided into 3 groups to submit the assessment answers using three channels - manual (paper based), Moodle and our workflow system. We collected stats (time duration in hours) for each task from - assessment creation to marks distribution back to students for all three mentioned channels. The tasks are described below:

- Question creation: Time taken to create assessment question.
- Student answers: Average time taken to answer all assessment questions.
- Answers collection: Approximate time taken to collect the student answers.
- Distribution among TAs: Approximate time taken to distribute student answers among TAs.
- TA Evaluation: Average time taken by TAs to evaluate student answers.
- Head TA consolidation: Time taken by Head TA to consolidate student answers from other TAs.
- Class distribution: Time taken by TAs to distribute evaluated student answers back to students.
- Students discussion: Time taken for evaluation discussion among TAs and students.
- Answers re-consolidation: Time taken by TAs consolidate student answers again after evaluation discussions.
- Marks consolidation: Time taken by Head TA to consolidate student marks in spread sheet or a system.
- Marks distribution: Average time taken by TAs to distribute marks to students.
- Total time duration: The total time taken to complete above mentioned 11 tasks sequentially.

Figure 2.4 shows a graph with time duration in hours for each of the task mentioned above, for channels - manual submission, Moodle submission and submission through our document workflow system. The graph shows (i) duration for each task - which is average time taken per task for all three channels of submission and (ii) total time duration - is the total time taken to assess students using three mentioned channels. We observed that, in general our document workflow system saves time for most

Class Room Experiment	count
No. of students	127
No. of questions	3
No. of instructors	4
Total answers	381

Table 2.1 Controlled Class Room Experiment details.

tasks as shown in the Figure 2.4. Our document workflow system also saves considerable time (average assessment time for class) when compared to manual handwritten paper based assessments. This is because few tasks can be skipped while using online assessments. As seen in Figure 2.4, the system also outperforms Moodle due to ease of use through mobile upload of assessments.

2.3.2 Discussion - Emerging Research Problems

2.3.2.1 Handwriting Plagiarism

Most universities use online plagiarism detection software to root out Internet plagiarism. The problem of predicting the similarity between two handwritten document images has already been addressed here [20, 21]. Though this is not a completely solved problem, we are trying to find better ways to enhance the ability to detect plagiarism among students. Our preliminary observations indicate that simple word spotting techniques does not suffice and we also need semantic techniques on handwritten text to solve the problem (Figure 2.5).

2.3.2.2 Author Identification Handwritten Text

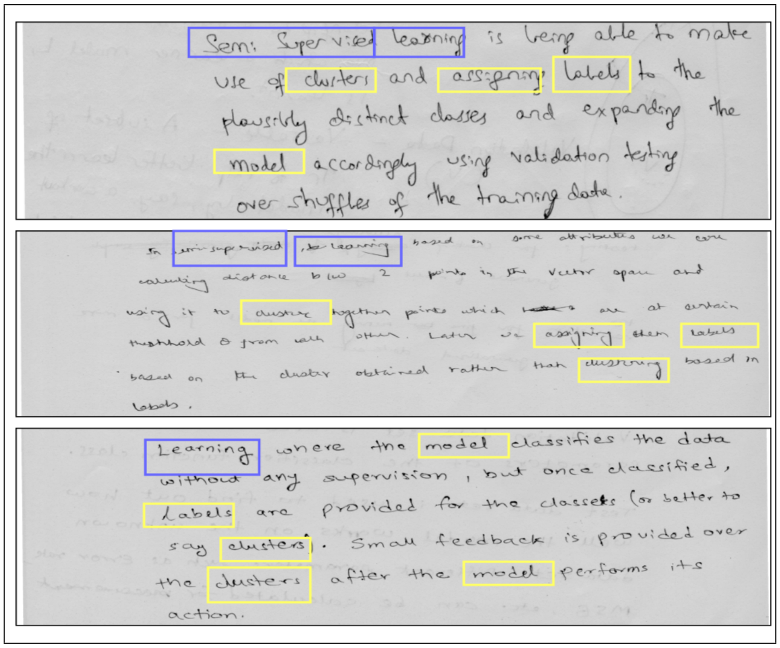
This is to identify documents containing more than one document signature style. A student typically spends several years in college. Hence a single document from student can used as unique fingerprint/signature to identify his handwriting across semesters. Our current module developed using method described in [22] is able to identify the students with decent accuracy but is not perfect. Better and faster methods are required to enhance both accuracy and speed when comparing across thousands of students on college premises.

2.3.2.3 Code Plagiarism

Plagiarism is a statement that someone copied code deliberately without attribution While MOSS [23] automatically detects program similarity, it has no way of knowing why codes are similar. Systems like MOSS also use web-services for code comparison which makes them even more slow. It is still up to a human to go and look at the parts of the code that MOSS highlights and make a decision about whether there is plagiarism or not. Though we have integrated a custom code analyzer which uses sequence based models [24], it is limited to C language and better models are required to scale to large number of students.

2.3.2.4 Evaluation of Handwritten Assessments

The typical engineering homework assessment may involve sketches, formulas with special symbols, as well as calculation steps. The most time efficient way for students to do this work is by hand, on paper. The handwritten assessment of student will be available for further evaluation by instructors, either



Semi Supervised learning is being able to make
use of clusters and assigning labels to the
plausibly distinct classes and expanding the
model accordingly using validation testing
over shuffles of the training data.

In semi-supervised ,~~to~~ learning based on some attributes we are
calculating distance b/w 2 points in the vector space and
using it to cluster together points which ~~[illegible]~~ are at certain
threshold θ from each other. Later we assigning them labels
based on the cluster obtained rather than clustering based in
labels.

Learning where the model classifies the data
without any supervision, but once classified,
labels are provided for the classes (or better to
say clusters). Small feedback is provided over
the clusters after the model performs its
action.

Figure 2.5 Sample Hand written ML assessment analyzed for plagiarism. Blue and yellow bounding boxes show common and important words using which, a plagiarism score is calculated.

using on screen evaluation tools or semi/auto evaluation methods which are still research problems as explained below.

2.3.2.5 Semi-automated Evaluation

In a university setting, tutors are required to evaluate several students and thousands of answers at a time. This can be cumbersome and any assistance provided to the instructors which can increase the throughput of evaluations will be a value-add. Clustering based assessment techniques are available for text based assessments [25]. The method first trains a model on similarity metric between student responses, but then go on to use this metric to group responses into clusters and sub clusters. A similar method can be implemented for handwritten evaluations where segmented words can be clustered based on semantic similarity between students response and reference answer given by the instructor. Student responses can be queued from the clusters based on the similarity metric which can increase the throughput of evaluations. We call this semi-automated evaluation of handwritten assessments. Our method can currently detect key phrases in the assessment.

2.3.2.6 Fully Automated Evaluation

Automated evaluation of handwritten assessments can be seen as an extension to the above mentioned method, where assistance was restricted to clustering answers, queuing them and highlighting the keywords in assessments. This can be further enhanced provided that the reference answer is available. A regression model can be trained on a set of semantic word features [8] in visual space, which can predict an evaluation score similar to that of an instructor. The score may not be necessarily accurate but we feel that a nearest score with a confidence metric can boost the throughput of evaluations enormously. We are currently testing the efficiency of this method and it is yet to be integrated into the our document workflow system.

2.4 Summary

Handwriting recognition has not reached a state that can directly help with the scalability of automated evaluations. However, we argue that our work flow system can enhance the efficiency and quality of the assessments without the need of OHR. Our system presented in this chapter addresses the need for a tool to computerize the existing handwritten assessments at all levels of our education system. Through this chapter we tried to showcase the capabilities of our document workflow system. To summarize, it has useful set of tools which encompass existing technologies for text, code and handwritten assessments, which can enhance the tutors and students experience alike by minimizing the time required for the whole assessment management process. Though the process is not yet perfect, the platform is open for future enhancements not only in text and handwritten work space but also in integrating research output from audio and video space. Automated evaluation of handwritten assessments is one such enhancement which can be useful to education system and can be integrated into our platform. In the next chapters, we discuss such enhancements which can help in bringing scalability in education system.

Chapter 3

POS Tagging and Named Entity Recognition on Handwritten Documents

Information extraction from handwritten document images has numerous applications, especially in digitization of archived handwritten documents, assessing patient medical records and automated evaluation of student handwritten assessments, to mention a few. Document categorization and targeted information extraction from various such sources can help in designing better search and retrieval systems for handwritten document images. Important studies have been undertaken on analysis of layout, printed and handwritten text separation and text/non-text segregation in documents. Moreover, better accuracy with higher efficiency has been achieved on pre-processing modules such as text-line identification, word/character segmentation, as well as Optical Character Recognition (OCR) engines. On degraded documents, where OCR engines do not work well, keyword spotting can play a remarkable role in identifying important words. Keyword spotting [26] is used for automatic document categorization by detecting the keywords or named entities directly on handwritten document images rather than transcribing to text to find keywords.

Semantic annotation of handwritten documents, especially spotting keywords using POS tags or NER is relatively a newer problem with very few works emerging on this front. POS tagging and Named Entity Recognition has been a prominent research work area in the field on Natural Language Processing (NLP) and Information Retrieval (IR) for last two decades. However, works on POS tagging and NE identification from document images are rare. In this chapter, we attempt to fill the gap by proposing an approach for POS tagging and NER without handwriting transcription. The contribution of this work is to show generalization with a similar or improved performance of a unified end-to-end model without separating the sequence of sub-processes involved, thereby avoiding error propagation. Identifying named entities using noun phrases from POS tags can also be greatly helpful for keyword-based document retrieval. Detecting named entities irrespective of its structural and positional characteristics (e.g. uppercase or lowercase letters) is an advantage of our approach. As a pre-processing step, we choose a handwritten dataset with segmented words and POS tag annotations. It helped us focus only on the aspect of POS and named entity tagging on handwritten word images rather than the problem of word segmentation from handwritten documents.

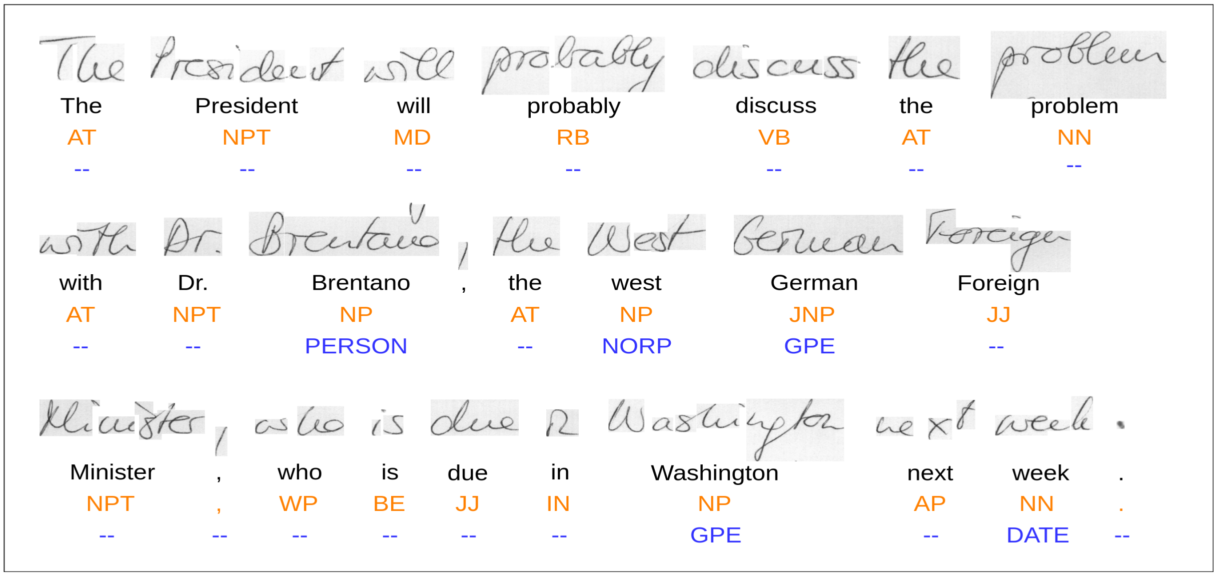

The President will probably discuss the problem
The President will probably discuss the problem
AT NPT MD RB VB AT NN
-- -- -- -- -- -- --
with Dr. Brentano , the West German Foreign
with Dr. Brentano , the west German Foreign
AT NPT NP AT NP JNP JJ
-- -- PERSON -- NORP GPE --
Minister , who is due in Washington next week .
Minister , who is due in Washington next week .
NPT , WP BE JJ IN NP AP NN .
-- -- -- -- -- -- GPE -- DATE --

Figure 3.1 Example of POS and NE tagging on a sentence chosen from IAM handwritten dataset.

3.1 Related Works

To the best of our knowledge, this is first time POS tagging has been attempted on Handwritten documents. Several state-of-the-art NER techniques were published in the literature using handcrafted features [27, 28]. However, work on POS tagging and Named Entity Recognition (NER) from handwritten document images is rare. Zhu et al. [29] discussed an approach for extracting relevant named entities from document images by combining rich page layout features in the image space with language content in the OCR text using a discriminator conditional random field model. They also employed an OCR engine and recognized the named entities with assistance from the OCR outputs.

One of the options is to transcribe and detect the named entities at the same time. The method described in [30] uses Hidden Markov Models and category n-grams to transcribe and detect categories in demographic documents, obtaining a quite good accuracy. However, the method is following a handwriting recognition architecture, and thus it depends on the performance of the optical model. It needs sufficient training data, and it is unable to detect or recognize out-of-vocabulary (OOV) words.

Toledo's [31] approach is based on Convolutional Neural Networks with a Spatial Pyramid Pooling layer to deal with the different shapes of the input images. However, they did not explain the effect of sequential information in words, where named entities can depend on relational positions of other entities. The ICDAR 2017 Information Extraction competition papers [32, 33], describe jointly training handwritten text recognition (HTR) and named entity recognition (NER), without separating them as subsequent tasks to mitigate the disadvantage of errors in the first module affecting the performance of the second module. In historical handwritten documents, handwriting recognition struggles to produce an accurate transcription thereby reducing the accuracy of the whole system.

Transcription based models such as [30, 32, 33] trained Handwritten Text Recognition (HTR) and NER jointly, to mitigate the disadvantage of errors in the first module affecting the next. But in historical handwritten documents, handwriting recognition struggles to produce an accurate transcription thereby reducing the accuracy of the whole system. Adak et al. [34] described an approach to directly detect the named entities from the document images. They used handcrafted features from document images with LSTM classifier, thereby avoiding the transcription step. The method relies on handcrafted features like identifying capital letters to detect possible named entities.

3.2 Our Approach

We hypothesize that, with sufficient handwritten document data and pre-processing, a deep learning model will be able to predict POS tags and named entities despite the inherent complexity, without the need for transcription.

3.2.1 Direct Learning using Synthetic Dataset

Deep learning architectures need large datasets to attain decent results on image recognition tasks and finding sufficient handwritten document images is a challenging task. Hence we first trained the model on synthetic handwritten word images. We used a standard parts-of-speech dataset to create a synthetic handwritten dataset using artificial fonts, as described in [20]. We used the same font for each sentence and sufficient data augmentation in the form of noise, translation, and rotation to resemble a large real handwritten dataset. Our assumption is that, with sufficient data, a deep learning model can generalize well on the end-to-end task without breaking it into sub-tasks [35]. For POS tagging on handwritten text, our first step was to choose a model trained on word spotting in handwritten document images. The use of deep learning architectures to capture spatial features of word images is widely discussed in [20, 36]. The authors used HWNet architecture trained on 1 million word image dataset to make it robust to most handwriting variations. We initially used the pre-trained model (HWNet) to extract the features of synthetic handwritten words and, later fine-tuned a separate neural net on these features to classify POS tags. We considered this model was our baseline for the best performance that can be achieved using a pre-trained model on handwritten word images. In our alternate training scheme, we directly train a deep model on word images to classify POS tags. We observed that the model performance was similar to HWNet feature-based model which affirmed our assumption that translation into text or feature extraction sub-tasks may not be required for POS tagging on handwritten word images.

3.2.2 POS Tagging and NER

The model trained on the synthetic dataset is fine-tuned on a real handwritten dataset. We tested various architectures (CNN, CNN-LSTM) for both POS tagging and NER on a challenging handwritten document dataset. Some of them are discussed below.

3.2.2.1 Deep CNN Model for POS Tagging

Convolutional Neural Nets (CNN) are good in capturing the intricate details of images, hence making the model stable to inconsistencies like noise and translation [37]. We trained a ResNet [38] model with 35 layers (validated empirically) on the synthetic dataset and fine-tuned it on IAM dataset for POS tagging task. The ResNet-35 ends with a softmax layer that outputs the probability distribution over the class labels (POS tags). We trained the model with cross-entropy loss function to predict the class labels.

3.2.2.2 CNN-LSTM Model for POS Tagging

The probability of a transition between words may depend not only on the current observation, but also on past and future observations, if available [39]. Since sentences in handwritten document images are word image sequences, we next used a combination of ResNet (CNN) and LSTM layers for training a POS tagging model on sequential information. We appended two layers of LSTM after ResNet-35 blocks and converted the input to LSTM as time distributed sequence. Different sequence lengths were tested on POS tags (classes). We report the performance of changing sequence lengths in the results section.

3.2.2.3 Named Entity Recognition

We adapt the similar architectures (CNN, CNN+LSTM) for the problem of NER. Here the underlying CNN architecture is ResNet-35. However, neither of the models had higher accuracy as noticed in similar experiments reported in [31]. We observed that named entities are related to position and distribution of POS tags in a sentence. We trained a multi-output classification network with architecture similar to POS model, with an extra branch of dense layers from the first fully connected dense layer, for named entity prediction. Hence the model now has an independent output with loss calculated from two sets of classes. As described in Section 3.3.2, named entities have class imbalance problem. This is one of the reasons for choosing outputs separated by multiple dense layers rather than a common layer training for multi-class classification. We initially trained the network simultaneously for both POS and NER.

Named Entities	**Tags**
Date	DATE
Geopolitical Entity	GPE
Organization	ORG
Person Name	PERSON
Nationalities or Religious or Political Groups	NORP
Unrelated	OTHERS
Not an Entity	–

Table 3.1 Named Entities used for our analysis. We chose 6 most commonly used named entities out of the 17 tags provided by Spacy tool.

Experiments	Precision	Recall	F1-score
Neural Net trained on HWNet features - CoNLL-2000 dataset synthetic images (POS tagging).	92.4	87.2	89.7
ResNet trained on - CoNLL-2000 dataset synthetic images (POS tagging).	94.2	84.5	89
ResNet trained on - CoNLL-2000 dataset synthetic images and fine-tuned on IAM dataset (POS tagging).	75.4	64.8	69.7
ResNet + LSTM trained on - CoNLL-2000 dataset synthetic images and fine-tuned on IAM dataset (POS tagging).	76.2	66.8	**71.2**
ResNet + LSTM trained on - CoNLL-2000 dataset synthetic images and fine-tuned on IAM dataset (NER).	74	64.1	**68.7**

Table 3.2 List of conducted experiments with precision, recall and F1-scores. We begin with basic model trained on synthetic dataset and end with a complex model (CNN + RNN) fine-tuned on handwritten dataset.

We observed that though POS prediction accuracy remained the same as independent POS training, NER training did not give encouraging results. Hence we first trained the model (ResNet + LSTM + dense layers) for POS tagging by freezing the dense layers of NER. After the network achieved satisfactory accuracy on POS tagging, we froze the POS part of the network - including the ResNet-LSTM layers and trained just the dense layers of NER. We used altered class weights to tackle the class imbalance problem. This method improved the accuracy of NER better than any of the methods we have tried earlier.

3.3 Experimental Results and Discussions

3.3.1 Dataset

We used two different datasets, for training and fine-tuning the models. For training, a synthetic handwritten dataset was generated from chunking dataset of CoNLL-2000 shared task [40], randomly using some of the 100 publicly available handwritten fonts [20]. The chunking dataset contains sentences aligned with 211727 text tokens along with their POS tags in a separate train and test text files. This dataset was initially used for training and validation. The model is further fine-tuned on IAM handwritten dataset [41]. The IAM dataset contains 1539 forms written by 657 authors. The forms are further segmented into 115320 words and are annotated with POS tags. Though IAM dataset contains segmented lines and sentences, they are not properly annotated with text accordingly which makes it difficult to demarcate the individual sentences accurately. Hence we separated sentences based on predefined sentence rules based on words and cross-validated them using python based NLP tool named "Spacy".

Since the IAM handwritten forms have transcripts, the text was fed into the Spacy for generating the ground truth named entities. Spacy tagged sentences with 17 different categories of named entities.

Though we restricted the classes to 6 named entities by choosing most recurrent tags, there was a class-imbalance problem. The list of tags used in this work is shown in Table 3.1. The unrelated entities occupied 92% of the NER classes. The IAM dataset is available as train, validation1, validation2, and test partitions. We used the training set to fine-tune our models and validated them against validation1 and validation2 sets.

3.3.2 Results and Discussion

As a baseline on the synthetic dataset, we initially extracted HWNet features on word images from the fully connected layer and trained a multi-layered perceptron on 36 POS classes provided by CoNLL-2000 dataset. The model achieved an F1-score of 89.7. We then trained a 35 layer ResNet model which achieved an F1-score of 89. This was our initial experiment to prove that a model can be trained to classify POS tags directly on handwritten word images, rather than a feature based model training.

3.3.2.1 POS Tagging on IAM Dataset

The ResNet model trained and validated on the synthetic CoNLL-2000 dataset is fined tuned on IAM dataset. We initially trained directly on word images to classify 58 POS tags without the sequence information. The architecture essentially contained no LSTM layers. The ResNet model achieved an F1-score of 69.7 on IAM test dataset. We altered the architecture and dataset to include sequence information. We replaced dense layers succeeding the CNN layers with LSTM layers and trained the model with varying sequence lengths of 3, 64, 128 and 256 words. We observed that ResNet-LSTM model trained on 128 word length sequences performed best with an F1-score of 71.2 We attribute the decline of prediction accuracy on IAM dataset compared to synthetic dataset due to the following reasons. (i) Distortions in word images - We observed that most of the word images are formed by concatenating individual characters. (ii) Character distortions - characters such as '.' and ',' are displayed as 'l' in the dataset. (iii) Proper nouns errors - proper nouns do not start with capital letters. We also observed that 26% of errors were due to the noun form of words (NN), followed by adjectives (JJ) at 18% and conjunctions (IN, TO) at 12%. Rest of the errors were due to special characters, commas, and full stops.

3.3.2.2 NER on IAM Dataset

Our models, training methods and metrics are summarized in Table 3.2. We used class weights to bias the training towards named entity tags other than "unrelated" class, to handle class imbalance problem. We initially trained IAM dataset words for two tasks in parallel using the architecture described in Section 3.2.2. But the accuracy of such model was low on NER task. Our first observation was that the errors caused by class imbalance were propagated back to the complete model which impacted the performance of both POS tagging and NER as well. Hence we first trained the model on POS tagging by freezing the NER layers, then we froze the layers for POS tagging and trained the model on NER.

After 20 epochs, we fine-tuned the whole model further using very low learning rate for 10 epochs. The ResNet-LSTM model gave F1-score of 68.7 on NER on handwritten text.

3.4 Summary

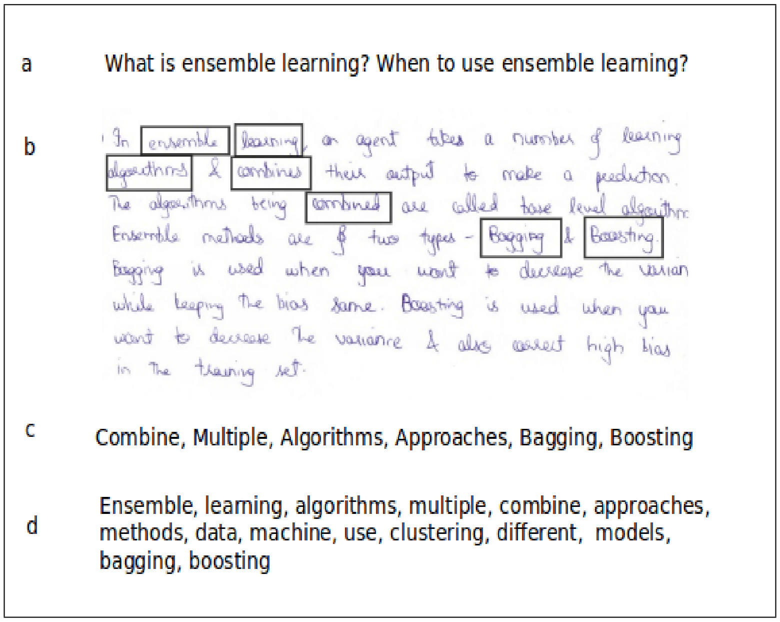
A POS tagger and named entity recognizer for offline handwritten unstructured documents, without employing a character/word recognizer and an independent linguistic model, is presented in this chapter. Experiments conducted on IAM dataset have resulted in an average F1-score of 71% on POS tagging and 68% on NER task. The proposed method is expected to work in other languages as well since our method deals with the linguistic aspect of handwritten documents where POS tags are identified first and then the NER. Our future work will endeavor to make our system more accurate for English scripts, where we can further restrict the POS tags to comply with PENN tree bank tags. In the next chapter, we put this feature to use in automating the evaluation of handwritten assessments.

Chapter 4

Towards Automated Evaluation of Handwritten Assessments

Scalable and reliable methods for evaluating the student performances are critically lacking in today's massive virtual as well as large real class rooms. As a result, instructors have to resort to simple boolean or multiple choice questions. It is well known that the handwritten responses are the most reliable means to evaluate the comprehension levels and the expressive skills of the students. They also reflect the students traits (e.g. concentration, logical organization of the thoughts, etc.) in a useful manner. Evaluating large numbers of handwritten answers is a time consuming, monotonous and costly task. An effective automatic evaluation system can contribute a lot to the teaching/learning process in different ways. Such a solution can prune the answers from a large class to a smaller number, and use the limited human resources judiciously. Even a minimal support like keyword highlighting can speed up the evaluation task. In an automatic setting, such a solution should reliably rank the answers. In this work, we are interested in designing a solution that helps in automatic evaluation of the answers as illustrated in Figure 4.1.

Automated evaluation of answers is an active area of research in the text domain. A multitude of measures for computing similarity between the true answer and the candidate answer have been proposed in the past based on surface level and semantic content features [3, 4]. Various linguistic aspects of the sentences were covered using WordNet [5], corpus-based features [6], Word2Vec [7], alignment-based features [8] and literal-based features [3]. Commercial systems (e.g. [9]) use a combination of statistical and natural language processing (NLP) techniques to extract linguistic features and use them in comparing answers. Though the text based automatic evaluation is nearing the reliable deployment in the university education system, handwritten answers are not yet amenable for their processing. Evaluation of handwritten answers needs significant advance in computer vision algorithms (e.g. text segmentation and recognition). A natural direction to evaluate the handwritten answers is to recognize the textual content and then exploit the advances in the text based automatic evaluation. While the printed text can be reliably recognized with optical character recognizer (OCR), offline handwritten text recognizer for unconstrained vocabulary are not robust enough for the practical use due to the inherent complexity of a handwritten word image.



a What is ensemble learning? When to use ensemble leaming?

b In ensemble learning, an agent takes a number of learning
algorithms & combines their output to make a prediction.
The algorithms being combined are called base level algorithm
Ensemble methods are of two types - Bagging & Boosting.
Bagging is used when you want to decrease the varian
while keeping the bias same. Boosting is used when you
want to decrease the variance & also correct high bias
in the training set.

c Combine, Multiple, Algorithms, Approaches, Bagging, Boosting

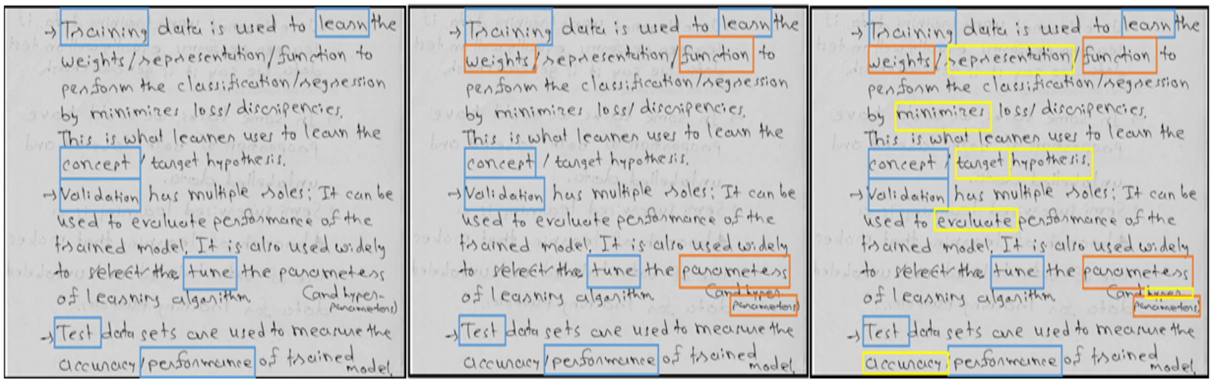
d Ensemble, leaming, algorithms, multiple, combine, approaches, methods, data, machine, use, clustering, different, models, bagging, boosting

Figure 4.1 A sample answer. a) Question from university exam, b) student's handwritten answer with word spotting, c) keywords from textual sample answer, and d) keywords after query expansion.

However, one can resort to image based matching methods (popularly known as word spotting [42]) for matching the textual content. More recently, with the popularization of deep architectures [36, 43, 21] and introduction of synthetic data [20] for training, there has been a significant improvement in both recognition and word spotting in multi-writer handwritten documents. In this work, we capitalize this success of deep features and develop our automatic handwritten evaluation framework. There has been only fewer attempts to address the problem of handwritten text assessments. Srihari [44] proposed a method for automatic scoring of short essays from reading comprehension tests. They presented an end to end pipeline with handwriting recognition, contextual post processing based on trigrams and evaluated scoring methods using a latent semantic analyzer and a trained neural network. Other attempts [45] in this space are also restricted to handwritten comprehensions with semi supervised evaluation and does not discuss much about word spotting with context analysis and how synonymy and polysemy are handled.

Measuring similarity of segments of text, works poorly with traditional document similarity measures based on word spotting (e.g., cosine), since there are often few terms in common between the two text snippets. Concepts such as "United Nations Secretary General" and "Ban Ki-Moon" should have high degree of semantic similarity and "United States" and "US" should refer to same named entity. These issues can not be addressed using the traditional word spotting of keyword which only captures the content and does not operate on the semantic information.

In this work, we limit our attention to evaluate the handwritten answers that are digitized as images. Our use case is an online system where students upload the handwritten answers as images digitized by their mobile phones or a scanner. We evaluate the appropriateness of the textual content for being the



→ Training data is used to learn the weights/representation/function to perform the classification/regression by minimizes loss/discripencies. This is what learner uses to learn the concept / target hypothesis.

→ Validation has multiple roles: It can be used to evaluate performance of the trained model. It is also used widely to ~~select the~~ tune the parameters of learning algorithm. (and hyper-parameters)

→ Test data sets are used to measure the accuracy/performance of trained model.

Figure 4.2 Samples of word spotting improvements with our context retrieval enhancements. i) Word spotting with ground truth keywords, ii) with query expansion, and iii) LDA with query expansion. We observe improvement of number of keywords spotted for question "How are training, validation and testing datasets useful in machine?"

answer to a given question automatically. Our method takes care of the natural variations in the answers, adapts word spotting to the course changes using self supervised learning and provide a reliable score that compares with the human evaluation. Towards this, we borrow ideas from information retrieval, document image analysis and feature based automated assessment. In the rest of the chapter, we present (i) a word spotting based automatic evaluation solution based on the deep learned features (Section 4.1). (ii) a self-supervised enhancement of the word spotting (Section 4.2.1). (iii) a set of features in the image space that captures the semantics and scores computable in the image space (Section 4.2.4) and (iv) experimental validation on a set of student answers from a real classroom (Section 4.3).

4.1 Scoring by Word Spotting in Images

We developed our scoring model based on a word spotting. Here, our interest lies in finding the matching score between the keywords associated with the Textual Reference Answer (TRA) and Handwritten (HW) document images, written by different writers in an unconstrained setting. Word spotting is typically formulated as a retrieval problem where the query is an exemplar image (query-by-example), and the task is to retrieve all word images with similar content. It uses a holistic word image representation which does not demand character level segmentation. Many of the popular features [46] are limited for the multiple-writer scenarios due to high intra-class variations. Such a problem is now successfully addressed using CNN features [20, 47] for handwritten word images. In this work, we used architecture inspired by HWNet-v2 [48] which is pre-trained on a large corpus of synthetic handwritten word images and later fine-tuned on IAM dataset [41]. The HWNet v2 is a ResNet34 network with 4 ResNet blocks and two fully connected (FC) layers as penultimate layers instead of global average pooling, as proposed in original ResNet architecture [38]. The model is further fine-tuned on the training datasets created by us (Section 4.3.1) to learn the natural variations in writer styles.

4.1.1 Keyword Extraction

A primary source of keywords for word spotting is the Textual Reference Answer (TRA) provided by instructors for each question. Keywords are either manually annotated by the examiner from TRA or extracted from TRA using NLP techniques. From the linguistic aspect, the building blocks of a sentence is a noun phrase (NP) and a verb phrase (VP). NP represents topics or subjects/objects in a sentence, while VP describe some action between the subject/objects in a sentence. We used the keywords from both NP and VP since they can sufficiently describe the topic and hence the context is derived from them. We used Stanford core NLP tools [49] like POS tagger and sentence parser to extract keywords from textual reference answer. The keywords thus extracted are further filtered by the examiner by intuition and experience, if required.

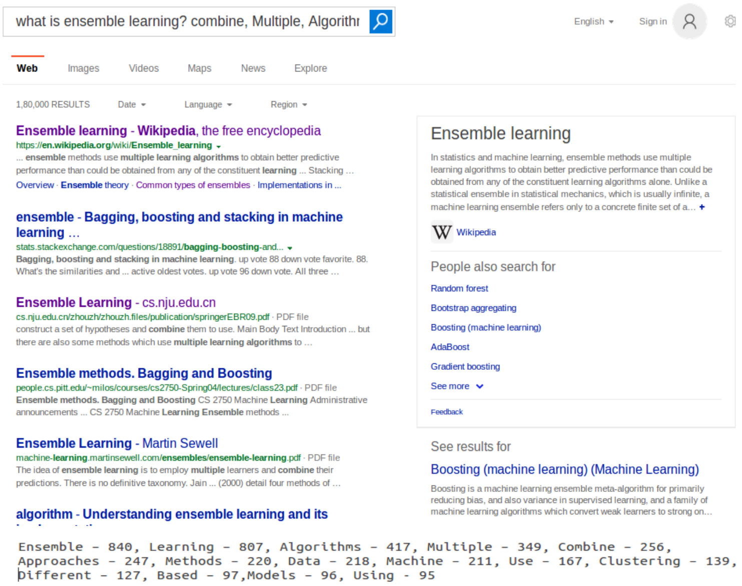
We match the keywords in the image space. For image matching, keywords from TRA are synthesized into images using multiple synthetic fonts. Given the keyword images, we extract the corresponding features from a model trained on word spotting. Later, we perform word spotting on segmented answer images from answer sheets using nearest neighbor search with a threshold set empirically. We observed that our model performs with an accuracy of 82% on our dataset (more details later).

Although the performance seems reasonable, we show in the next section that given the nature of our problem, we can further improve the word spotting performance by restricting the vocabulary to a particular domain. A grading framework solely dependent on keywords from textual reference answer would be unable to detect semantically relevant keywords, thus marking multiple answers invalid. Figure 4.1 demonstrates an example of a handwritten answer with just the reference answer based keywords and semantically related keywords. In the next section, we present our enhancements to address these issues.

4.2 Enhancements

4.2.1 Self Supervised Word Spotting

It is a well-known fact that CNN trained for a related task could be adapted or fine-tuned to get reasonable and even state-of-the-art performance for new tasks [50]. In our case, we use a similar strategy where we reformulate the problem of word spotting from generic vocabulary to word classification limited to question/reference answer specific keywords. While grading a specific question, we are interested in doing accurate word spotting only on a set of words that are semantically related to the TRA (discussed in Section 4.2.2). Since the domain of keywords for a specific question is limited (approximately 5-25 words), we fine-tune the model to spot these limited keywords more accurately. We froze all the layers of the model (discussed in Section 4.1) except the FC layers, replacing softmax layer to match the number of new keywords and fine-tune the model with very low learning rate. For generating the training data automatically from the keywords of TRA, we use synthetic handwritten fonts as suggested in [48]. This process repeats for every new question and its reference answer (TRA). We



Ensemble learning - Wikipedia, the free encyclopedia
https://**en.wikipedia.org**/wiki/**Ensemble_learning**
... **ensemble** methods use **multiple learning algorithms** to obtain better predictive performance than could be obtained from any of the constituent **learning** ... Stacking ...
Overview · **Ensemble** theory · Common types of ensembles · Implementations in ...

ensemble - Bagging, boosting and stacking in machine learning ...
stats.stackexchange.com/questions/18891/**bagging-boosting**-and...
Bagging, boosting and stacking in machine learning. up vote 88 down vote favorite. 88. What's the similarities and ... active oldest votes. up vote 96 down vote. All three ...

Ensemble Learning - cs.nju.edu.cn
cs.nju.edu.cn/zhouzh/zhouzh.files/publication/springerEBR09.pdf · PDF file
construct a set of hypotheses and **combine** them to use. Main Body Text Introduction ... but there are also some methods which use **multiple learning algorithms** to ...

Ensemble methods. Bagging and Boosting
people.cs.pitt.edu/~milos/courses/cs2750-Spring04/lectures/class23.pdf · PDF file
Ensemble methods. Bagging and Boosting CS 2750 Machine **Learning** Administrative announcements ... CS 2750 Machine **Learning Ensemble** methods ...

Ensemble Learning - Martin Sewell
machine-**learning**.martinsewell.com/**ensembles**/**ensemble-learning**.pdf · PDF file
The idea of **ensemble learning** is to employ **multiple** learners and **combine** their predictions. There is no definitive taxonomy. Jain ... (2000) detail four methods of ...

algorithm - Understanding ensemble learning and its

Ensemble learning

In statistics and machine learning, ensemble methods use multiple learning algorithms to obtain better predictive performance than could be obtained from any of the constituent learning algorithms alone. Unlike a statistical ensemble in statistical mechanics, which is usually infinite, a machine learning ensemble refers only to a concrete finite set of a... +

W Wikipedia

People also search for

Random forest
Bootstrap aggregating
Boosting (machine learning)
AdaBoost
Gradient boosting
See more

Feedback

See results for

Boosting (machine learning) (Machine Learning)

Boosting is a machine learning ensemble meta-algorithm for primarily reducing bias, and also variance in supervised learning, and a family of machine learning algorithms which convert weak learners to strong on...

```
Ensemble - 840, Learning - 807, Algorithms - 417, Multiple - 349, Combine - 256,
Approaches - 247, Methods - 220, Data - 218, Machine - 211, Use - 167, Clustering - 139,
Different - 127, Based - 97,Models - 96, Using - 95
```

Figure 4.3 Example of results obtained from querying the search engine. We can observe contextually relevant terms in definitions along with query terms.

refer this as self-supervised word spotting where the entire process happens without any external human supervision.

4.2.2 Contextual Query Expansion

Word spotting using keywords from TRA provides baseline scores for the evaluation. However, students are likely to use paraphrasing with synonyms and acronyms in answers which can make automatic evaluations difficult. Alternatively, we can expand keywords using knowledge-based sources like WordNet and Thesauri but can result in false positives due to underlying ambiguity in word senses which could be only resolved by understanding the context. Other sources like Wikipedia articles, query reformulation logs and search results obtained from the web (together called as corpus-based sources) provides a set of contextual texts that are used to expand the original sparse keyword representation [51]. In our experiments, we use web search results to expand our query representations.

Query expansion is formulating a given query to retrieve a relevant document or information retrieval. It involves finding various semantically related words from words in a query such as synonyms, antonyms, meronyms, hyponyms, and hypernyms. It also involves a pre-processing step of stemming the queried words and automatically fixing the spelling errors. We observed that the keywords embedded in a question and textual reference answer could help in understanding the context and hence narrow down extraction of contextually relevant information significantly. We run constructed query of words against a Bing search engine's index and retrieve the top 500 documents [52]. The titles and descriptions from results are then concatenated and used as our expanded keyword representation.

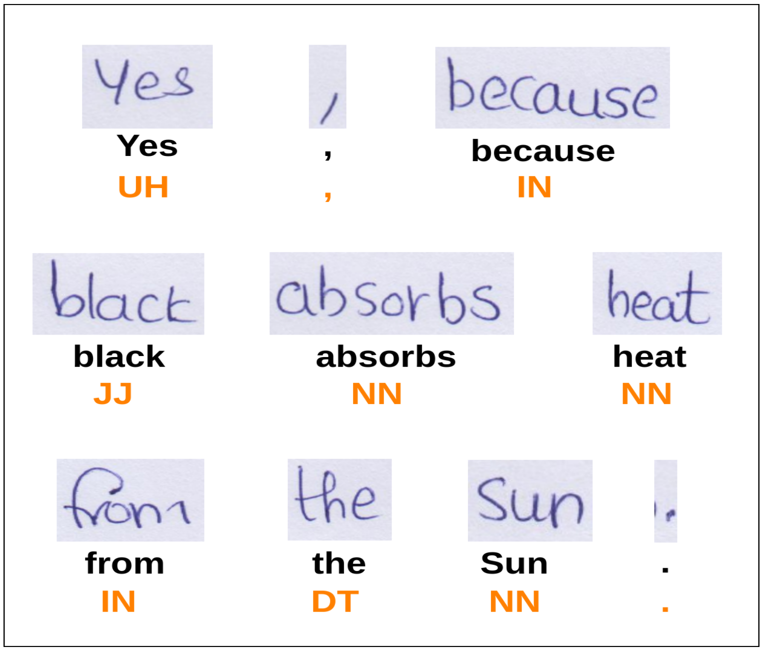


Figure 4.4 The figure shows an example from the SE dataset where words are classified into POS tags. Word images with the transcribed text and their POS tags are available during train and testing.

In Figure 4.3, we show portion of the expanded representation for the short text segment "ensemble learning". As we see, this expanded representation has many contextually relevant terms, such as "Bagging", "Boosting" and "AdaBoost" that are not present in the surface keyword representation. To pick the most informative keywords from these results, we first weight each expanded keywords using TF-IDF scores and select only the top-N words. In another approach, we considered a query as "topic" and Latent Dirichlet Allocation (LDA) [53] is used on query results (documents) to form a cluster of words that often occur together. Using contextual clues, topic models can connect words with similar meanings and distinguish between uses of words with multiple meanings. We used MALLET framework [54] for topic modeling.

4.2.3 POS Tagging and NER

Despite the usage of a good semantic query expansion methods, we may not to retrieve the necessary keywords every time. Since we are working on automated short answer evaluation, keywords are not always relevant. Boolean answers are not uncommon in assessments and at times an adverb like "not" can change the meaning of the answer despite presence of keywords. Hence, parts of speech (POS) tagging and named entity recognition (NER) on handwritten document images are helpful as an extra set of features for automated evaluation. POS and NER tagging is a NLP problem, to parse a sentence and assign parts-of-speech tags per word and classify the words into pre-defined entity categories such as the names of people, streets, organizations, dates, etc. POS tagging and key phase detection (Figure 4.4) from document image is quite difficult without transcription to text. However, such detection is essential since handwritten text recognition is not yet perfected and hence NLP tools cannot be used directly [55]. We used POS tags and named entities spotted from the student's answers as additional features to model and automatically evaluate the student handwritten answers. For this, we used a method described in [56, 33, 57] where, a CNN + RNN model architecture is used to take the advantage of sequential knowledge in successive word images. We trained a similar architecture on IAM dataset to detect POS tags and named entities directly from word images segmented from the handwritten text without transcribing word images to text. We used 58 unique POS tags and 6 named entities obtained using python based NLP tool named Spacy for tagging on the datasets.

4.2.4 Features and Grading

Our aim is to design a solution that assigns a quantitative score that is very similar to the score assigned by a human instructor. We do this by training a neural network in a supervised way on a set of features described below.

4.2.4.1 Base Features

The keywords spotted from TRA in a student's handwritten answer is the essential clue of its proximity to the textual reference answer. We capture this with (i) ***unique terms***: the count of unique keywords from TRA spotted in the students answer. (ii) ***keyword recall***: the ratio of *unique terms* spotted to count of actual keywords in TRA and (iii) ***word count***: the number of words segmented from the text. We refer to these three features as the BASE FEATURES.

4.2.4.2 Lexical Features

We also capture the features related to the lexical complexity. They are (iv) ***tokens***: the total number of terms from the ground truth keywords (from TRA) spotted, including term repetitions This feature characterizes the student's domain vocabulary knowledge (and not the common words). These features are like noun phrases and repeated n-grams [4] captured by a parser on a transcribed text. (v) ***unique terms - token ratio***: the ratio of the number of the unique terms spotted, to that of *tokens* [9]. The purpose of this feature is to capture the excessive use of keywords to enlarge the answer artificially instead of the precise description.

4.2.4.3 Syntactic Features

We use the following features to capture the **syntactic clues** from the images using word spotting. (vi) ***words length***: a simple word count obtained after segmentation of handwritten answer image after filtering out anomalies based on word image size. (vii) ***term strength***: the purpose of this feature is to count the number of unique terms in the answer and standardize this count with the total number

Controlled	**Count**
No. of Students	15
No. of Questions	10
Total Answers	150
Class Room	
No. of Students	96
No. of Questions	6
Total Answers	576
SciEntsBank Handwritten	
No. of Students	12
No. of Questions	69
Total Answers	3152

Table 4.1 Details about the datasets used in our experiments - Controlled, Class Room and SciEntsBank.

of words in the essay. (viii) ***token strength***: the purpose of this feature is to count the tokens in the answer and standardize this count with the total number of words in the essay. It captures the strength of prioritized usage of the contextual words instead of simple words.

4.2.4.4 NLP Features

We capture the semantic clues by measuring the organization of the answers in terms of the presence of named entities and its supporting keywords in phrase or sentence. We used the method described in Section 4.2.3 to classify the words into their respective POS and named entity tags. We used the following features to capture the semantic clues. (ix) ***nouns phrase ratio***: ratio of nouns and adjectives spotted in students answer, with respect to nouns and adjectives in textual reference answer (TRA). (x) ***verb phrase ratio***: ratio of verbs and adverbs spotted in students answer, with respect to verbs and adverbs in textual reference answer (TRA). (xi) ***named entities match count***: total count of named entities matched between students answer and textual reference answer. Features described from (iv) to (xi) are together referred as SEMANTIC FEATURES.

With all these features computed from the student handwritten answers, we train a simple multi-layered neural network to predict the human score. We trained the network using mean squared error (MSE) loss and stochastic gradient descent (SGD optimizer to predict a score in the range $[0, 1]$.

4.3 Experiment Results and Discussion

4.3.1 Datasets

To validate our method, we collected handwritten answers to a set of questions from school and college students. We selected questions from three domains: machine learning, operating systems, and

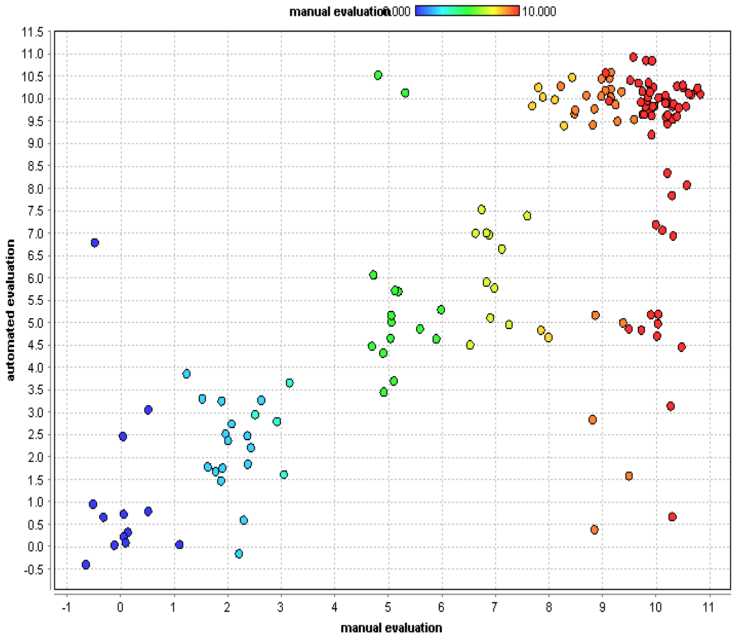


Figure 4.5 Comparison of average scores from manual evaluation (x-axis) and automatic evaluation (y-axis) for questions in CRD dataset. The scores are scaled till 10 for better plotting.

basic science. We choose these domains due to the matured vocabulary of these areas and presence of enough Internet resources. The questions are mostly descriptive, listing or differences based. Typical answers are one to four sentences long. Examples of questions in our dataset are: (a)“What are the roles of training, validation and test datasets in machine learning?” (b)“Why is dimensionality reduction is very popular in many machine learning solutions as a pre-processing step?” Examples of handwritten answers is shown in Figure 4. In all these cases, a human evaluated the answer first, and the human score is normalized to $[0, 1]$, and used as a signal for the supervision or the evaluation. We created corresponding textual reference answer and textual students answers separately for validation.

4.3.1.1 Class Room Dataset (CRD)

This dataset consists of answers from an actual university examination. We describe the details in Table 4.1. This dataset consists of a set of 6 questions answered by 96 students in an examination. The total number of answers extracted is 576. An independent human evaluator HE provided a score $[0, 1]$ based on the correctness of the answer.

4.3.1.2 Controlled Dataset (CD)

We created this dataset in an artificial class environment wherein 15 students participated to answer 10 questions. This dataset has simple questions, to imitate complexity of questions in high schools and colleges. As described in Table 4.1, we obtained a limited dataset of 150 answers from this exercise. This dataset have images, their corresponding text and the human scores.

4.3.1.3 SciEntsBank Dataset (SE)

The textual corpus was created as a part of Joint Student Response Analysis and Recognizing Textual Entailment Challenge in text domain [58]. The task is to develop models for automating the assessment of student responses to questions in the science domain. Of the two datasets provided, we used SciEntsBank Dataset (SE) for our third experiment, since this dataset contains a single reference answer provided by an expert instructor to every question and a clear demarcation in answer evaluation. The evaluation of datasets are given in three formats: i) 2-way, ii) 3-way and iii) 5-way evaluation schemes where labels focused on correctness and completeness of the response content. We evaluated student answers against the reference answer, using the 2-way evaluation scheme which classifies the answer either as “correct” or “incorrect”.

The SciEntsBank test corpus has about 5835 responses to 196 assessment questions in 15 different science domains. The test corpus is further divided into Unseen Answers (UA), Unseen Questions (UQ) and Unseen Domains (UD). We selected a subset of 69 questions from complete test corpus based on simplicity of answers and converted the corresponding multiple textual answers provided per question in the dataset, into 3152 handwritten student answers with the help of 12 students. We chose this dataset due to its relevance in the research community for ASAG task. This dataset also covers a broader domain of science and not just subject based question answers as in our earlier datasets.

4.3.2 Evaluation Methodology and Metrics

We quantitatively evaluated performance of the automatic evaluation (AE) exhaustively. The experiments do not consider the accuracy of segmentation in reporting evaluation metrics. We compare performance of our solution with that of human evaluation (HE) in the following way. First, we normalize the AE and the HE scores to a binary $[0, 1]$ value to reflect notation of “correct” and “incorrect” answers. Note that the AE and the HE scores are in the range of $[0, 1]$. Since even a low score from HE reflects certain degree of correctness in students answer, we lowered the threshold θ to 0.25 from 0.5 when converting to a binary range. Automatic evaluation is valid, if both the human and algorithm scores match. Otherwise, we consider AE as incorrect. We then compute, precision, recall and F1-score for the automatic evaluation.

4.3.3 Qualitative Results

We conducted 5 different experiments based on the keywords from TRA and keywords using different query expansion methods described in Section 4.2.2. These experiments were conducted first with BASE FEATURES and then with SEMANTIC FEATURES, as shown in Table 4.2 & 4.3. The first experiment **Base**

	CRD dataset			**CD dataset**			**SE dataset**		
Experiments	P	R	F1	P	R	F1	P	R	F1
Base Keywords	0.61	0.78	0.68	0.84	0.80	**0.82**	0.72	0.63	0.67
QE on Question	0.65	0.70	0.67	0.71	0.54	0.61	0.70	0.62	0.66
QE on Question & TRA	0.70	0.75	0.72	0.73	0.55	0.63	0.68	0.60	0.64
TF-IDF based QE	0.71	0.76	0.73	0.71	0.48	0.57	0.70	0.68	**0.69**
LDA based QE	0.71	0.78	**0.74**	0.70	0.51	0.59	0.71	0.65	0.68

Table 4.2 The table show results for all experiment methods using **base features** on CRD, CD and SE datasets. The experiments are listed on the left. QE stands for query expansion, P for precision, R for recall and F1 for F1-score.

	CRD dataset			**CD dataset**			**SE dataset**		
Experiments	P	R	F1	P	R	F1	P	R	F1
Base Keywords	0.67	0.79	0.72	0.64	0.75	0.69	0.63	0.72	0.67
QE on Question	0.65	0.69	0.67	0.64	0.73	0.68	0.64	0.75	0.69
QE on Question & TRA	0.70	0.72	0.71	0.63	0.79	**0.70**	0.66	0.64	0.65
TF-IDF based QE	0.69	0.85	**0.76**	0.71	0.60	0.65	0.66	0.75	0.70
LDA based QE	0.71	0.77	0.74	0.62	0.78	0.69	0.70	0.74	**0.72**

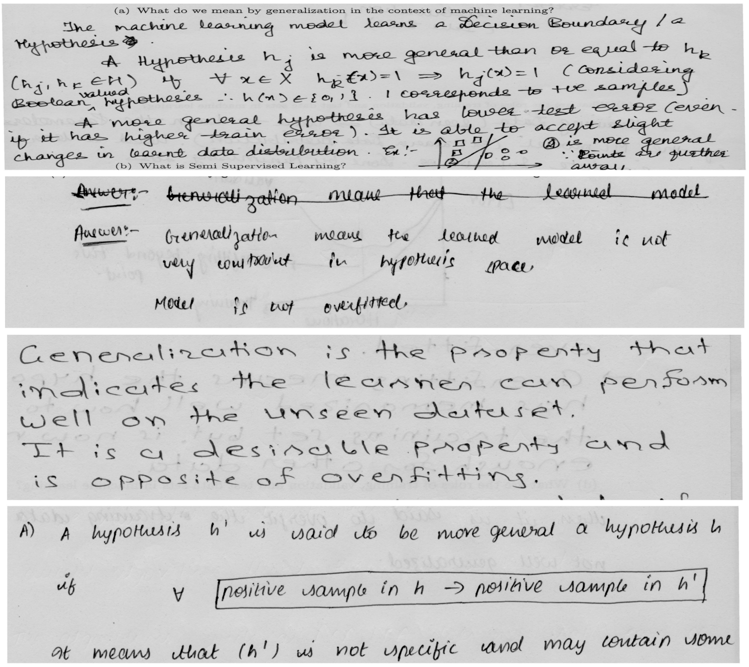
Table 4.3 The table show results for all experiment methods using **semantic features** on CRD, CD and SE datasets. The experiments list is on the left. QE stands for query expansion, P for precision, R for recall and F1 for F1-score.

Keywords was with keywords from TRA. In the second, both verb phrases and noun phrases extracted from the question are used in **Query Expansion on Question** experiment. This experiment sets the platform for unsupervised evaluation where keywords are from the question but not TRA, and therefore human intervention is not required fro creating a TRA.

We used bing search API to query the keywords from the question. The results were tokenized, converted to lower case, stop words were removed, and top 15 most repeating words were extracted and used as query words for word spotting. The model is trained on features obtained from the expanded representation. In the **Query Expansion on Question &** TRA experiment, the relevant query words are extracted from web, based on the keywords from both question and TRA. An example of query expansion is seen in Figure 4.3. Not all keywords are equally important in the context of a question. Hence, we performed a **Weighted Query Expansion** experiment with top-N keyword weights calculated from search result documents using TF-IDF scores. We conducted another experiment using LDA **based Query Expansion** from search results.

4.3.3.1 Base Features based Evaluation

We demonstrate the assessment performance using just BASE FEATURES obtained using the method described in Section 4.1. We used total word count, unique keyword count and keyword recall as features



(a) What do we mean by generalization in the context of machine learning?

The machine learning model learns a Decision Boundary / a Hypothesis.

A Hypothesis h_j is more general than or equal to h_k ($h_j, h_k \in H$) if $\forall x \in X$ $h_k(x)=1 \Rightarrow h_j(x)=1$ (Considering Boolean valued hypotheses $\therefore h(x) \in \{0,1\}$. 1 corresponds to +ve samples)

A more general hypotheses has lower test error (even if it has higher train error). It is able to accept slight changes in learnt data distribution. Ex:- ② is more general ∵ Points are further away.

(b) What is Semi Supervised Learning?

~~Answer:- Generalization means that the learned model~~

Answer:- Generalization means the learned model is not very contraint in hypothesis space.

Model is not overfitted.

Generalization is the property that indicates the learner can perform well on the unseen dataset. It is a desirable property and is opposite of overfitting.

A) A hypothesis h' is said to be more general a hypothesis h if $\forall$ $\boxed{\text{positive sample in } h \rightarrow \text{positive sample in } h'}$

it means that (h') is not specific and may contain some

Figure 4.6 List of failure scenarios due to i) figures and equations, ii) scratched lines, iii) improper word, character spacing and, iv) text highlighting using boxes.

for training and testing the model. Each dataset is split into training and testing sets, and we use the prediction from trained model to evaluate the answer as valid or invalid. Prediction probability, which is in the range of 0 and 1 is used as our grading score, as described in Section 4.3.2. From Table 4.2, we observe high precision scores across most of the experiments. We observed better performance using query expansion methods on CRD (using LDA) and SE (using TF-IDF) datasets, but CD dataset has a better score with base keywords. We attribute this due to presence of more definition and list-based questions in CD dataset, where keywords from TRA are sufficient and may not need query expansion. We observed that the baseline method perform poorly on the dataset of higher complexity (SE).

4.3.3.2 Semantic Features based Evaluation

In the second set of experiments, we added semantic features mentioned in Section 4.2.4 in addition to baseline features. From Table 4.3, it is evident that the accuracy of semantic features is better than base features for the complex CRD dataset. We also observed high recall scores across most of the experiments. We argue that this is probably due to combination of an increase in the number of features and keyword coverage by query expansion methods. From the Table 4.3, we observe better performance using query expansion (LDA specifically) methods on all the datasets. These experiments prove that topic modeller trained on search query documents and weighted query expansion methods (TF-IDF) has better key terms for word spotting.

We observed from above experiments proves that the automation (semi-supervised) in keyword extraction from the question and TRA using query expansion can help instructor with evaluation and grading. The results in Table 4.2 & 4.3 in general show that models trained on semantic features perform better than the base features and query expanded keywords provide better coverage of keywords for word spotting based evaluation.

4.3.4 Discussion

Our method is a pipeline integrating information retrieval and NLP based feature analysis. Errors in initial stages of document image analysis gets propagated and impact evaluation scores to a certain extent. A primary limitation of our work is the lack of comprehension of complex mathematical equations and inferences, as shown in Figure 4.6. Tidiness and organized answers also matter. Our prototype fails to segment text with less spacing between words, high skew and excessive word scribbling which are add up in word count thereby effecting scores. Answers paraphrased with simple non-technical terms were also found relatively hard to evaluate. However, we hope that our approach with some changes can address the grading requirements in a variety of subjects across domains.

4.4 Summary

We demonstrate an automatic evaluation scheme for handwritten answers with high correlation to the human evaluation. As a first step towards fully automating the grading schemes, we believe, this can act as an assistance to the instructors. Our framework integrates document image analysis, information retrieval, and feature based word spotting. On real answers from a classroom, it provides scores that correlate highly with the human evaluators. The method aimed at short descriptive answers, and it meets this purpose. With this chapter, we conclude the flow of experiments and applications specifically designed to bring scalability into the field of education. We focused our attention specially on handwritten document images generated in education system and build an extensible framework and an application which uses latest enhancements in both handwritten document analysis and deep learning.

Chapter 5

Conclusions and Future Directions

In this thesis we explored several areas in the domain of handwritten document analysis and applications. Each of the contribution has potential real world application. We have discussed the applications and future directions of our projects below.

In **Chapter 2**, we discussed a document image workflow system that helps in scaling the handwritten student assessments in a typical university setting. We demonstrated (i) a distributed image capture module with a mobile phone, (ii) image processing algorithms that improve the quality and readability, and (iii) image annotation module that process the evaluations/feedbacks as a separate layer.

- **Applications:** The platform provides useful set of tools which encompass existing technologies for text, code and handwritten assessments, which can enhance the tutors and students experience alike by minimizing the time required for the whole assessment management process. Though the process is not yet perfect, the platform is open for future enhancements not only in text and handwritten work space but also in integrating research output from audio and video space.

- **Future directions:** Future work involves implementation of information kiosk, an android based application with stylus support, for deploying on tablets. The user interface will support touch navigation to enhance the student assignment evaluation process. This can increase the throughput enormously.

In **Chapter 3**, we proposed an approach to detect POS and Named Entity tags directly from offline handwritten document images without explicit character/word recognition. This was based on the observation that POS tagging on handwritten text sequences increases the predictability of named entities and also brings a linguistic aspect to handwritten document analysis.

- **Applications:** The primary application of this work is in document image analysis and archival where named entities or key phrases detected by our method can be integrated into a pipeline of other applications such as document image indexing, automated evaluation of student answers and e-book tagging to mention a few.

- **Future directions:** The proposed method can be further enhanced to work in other languages as well since our method deals with the linguistic aspect of handwritten documents where POS tags are identified first and then the NER. Our future work will endeavor to make our system more accurate for English scripts, where we can further restrict the POS tags to comply with PENN tree bank tags.

Finally in **Chapter 4**, we describe an effective method for automatically evaluating the short descriptive handwritten answers from the digitized images. Our solution is based on the observation that a human evaluator judges the relevance of the answer using a set of keywords and their semantics. We modeled the problem as a self supervised, feature based classification problem, which can fine tune itself for each question without any explicit supervision.

- **Applications:** We believe, our work can act as an assistance to the instructors. Our framework integrate document image analysis, information retrieval and feature based word spotting. On real answers from classroom, it provides scores that correlates highly with the human evaluators. Our method is aimed at short descriptive answers and it meets this purpose.

- **Future directions:** Future work will focus on complete automation of evaluation of short answers without ground rules for feature extraction. We plan to implement successful scenarios seen in text based short answer grading system into handwritten evaluations, where word images can be converted to embeddings where words are related in semantic space.

Related Publications

1. Vijay Rowtula, Varun Bhargavan, Mohan Kumar, C.V. Jawahar. **Scaling Handwritten Student Assessments with a Document Image Workflow System**. Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops 2018, pages 2307–2314.

2. Vijay Rowtula, Praveen Krishnan, C.V. Jawahar. **POS Tagging and Named Entity Recognition on Handwritten Documents**. Proceedings of the 15th International Conference on Natural Language Processing (ICON-2018), pages 87–91.

3. Vijay Rowtula, Subba Reddy Oota, C.V. Jawahar. **Towards Automated Evaluation of Handwritten Assessments**. The Fifteen International Conference on Document Analysis and Recognition (ICDAR 2019).

Bibliography

[1] R. Milewski and V. Govindaraju. Handwriting analysis of pre-hospital care reports. In *CBMS*, 2004.

[2] S. N. Srihari. Handwritten address interpretation: a task of many pattern recognition problems. *International journal of pattern recognition and artificial intelligence*, 2000.

[3] D. Kanejiya, A. Kumar, and S. Prasad. Automatic evaluation of students' answers using syntactically enhanced lsa. In *Proceedings of the HLT-NAACL 03 Workshop on Building Educational Applications Using Natural Language Processing - Volume 2*, HLT-NAACL-EDUC '03, 2003.

[4] Y. Liu, C. Sun, L. Lin, X. Wang, and Y. Zhao. Computing semantic text similarity using rich features. In *PACLIC*, 2015.

[5] T. Wei, Y. Lu, H. Chang, Q. Zhou, and X. Bao. A semantic approach for text clustering using wordnet and lexical chains. *Expert Systems with Applications*, 2015.

[6] A. Islam and D. Inkpen. Semantic text similarity using corpus-based word similarity and string similarity. *ACM Transactions on Knowledge Discovery from Data (TKDD)*, 2008.

[7] T. Kenter and M. de Rijke. Short text similarity with word embeddings. ACM, 2015.

[8] M. Mohler, R. Bunescu, and R. Mihalcea. Learning to grade short answer questions using semantic similarity measures and dependency graph alignments. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies-Volume 1*. Association for Computational Linguistics, 2011.

[9] Y. Attali and J. Burstein. Automated essay scoring with e-rater® v. 2. *The Journal of Technology, Learning and Assessment*, 2006.

[10] S. Hepplestone, G. Holden, B. Irwin, H. J. Parkin, and L. Thorpe. Using technology to encourage student engagement with feedback: a literature review. *Research in Learning Technology*, 2011.

[11] K. Brandl. Are you ready to moodle. *Language Learning & Technology*, 2005.

[12] A. McAuley, B. Stewart, G. Siemens, and D. Cormier. The mooc model for digital practice. 2010.

[13] A. G. Picciano. Beyond student perceptions: Issues of interaction, presence, and performance in an online course. *Journal of Asynchronous learning networks*, 2002.

[14] C. Piech, J. Huang, Z. Chen, C. Do, A. Ng, and D. Koller. Tuned models of peer assessment in moocs. *arXiv preprint arXiv:1307.2579*, 2013.

[15] A. Majumdar, P. Krishnan, and C. Jawahar. Visual aesthetic analysis for handwritten document images. In *Frontiers in Handwriting Recognition (ICFHR), 2016 15th International Conference on*. IEEE, 2016.

[16] P. Ye, J. Kumar, L. Kang, and D. Doermann. Real-time no-reference image quality assessment based on filter learning. In *Computer Vision and Pattern Recognition (CVPR), 2013 IEEE Conference on*. IEEE, 2013.

[17] F. Shafait and T. M. Breuel. Document image dewarping contest. In *2nd Int. Workshop on Camera-Based Document Analysis and Recognition, Curitiba, Brazil*, 2007.

[18] K. Arvind, J. Kumar, and A. Ramakrishnan. Line removal and restoration of handwritten strokes. In *Conference on Computational Intelligence and Multimedia Applications, 2007. International Conference on*. IEEE, 2007.

[19] D. Merkel. Docker: lightweight linux containers for consistent development and deployment. *Linux Journal*, 2014.

[20] P. Krishnan and C. Jawahar. Matching handwritten document images. In *European Conference on Computer Vision*. Springer, 2016.

[21] S. Sudholt and G. A. Fink. Phocnet: A deep convolutional neural network for word spotting in handwritten documents. In *Frontiers in Handwriting Recognition (ICFHR), 2016 15th International Conference on*. IEEE, 2016.

[22] D. Bertolini, L. S. Oliveira, E. Justino, and R. Sabourin. Texture-based descriptors for writer identification and verification. *Expert Systems with Applications*, 2013.

[23] A. Aiken. Measure of software similarity. *URL http://www. cs. berkeley. edu/-aiken/moss. html*, 1994.

[24] J. Yasaswi, S. Purini, and C. Jawahar. Plagiarism detection in programming assignments using deep features. In *2017 4th IAPR Asian Conference on Pattern Recognition (ACPR)*, pages 652–657. IEEE, 2017.

[25] S. Basu, C. Jacobs, and L. Vanderwende. Powergrading: a clustering approach to amplify human effort for short answer grading. *Transactions of the Association for Computational Linguistics*, 2013.

[26] A. Fischer, A. Keller, V. Frinken, and H. Bunke. Lexicon-free handwritten word spotting using character hmms. *Pattern Recognition Letters*, 2012.

[27] A. Ritter, S. Clark, O. Etzioni, et al. Named entity recognition in tweets: an experimental study. In *Proceedings of the conference on Empirical Methods in Natural Language Processing*. Association for Computational Linguistics, 2011.

[28] G. Lample, M. Ballesteros, S. Subramanian, K. Kawakami, and C. Dyer. Neural architectures for named entity recognition. *arXiv preprint arXiv:1603.01360*, 2016.

[29] G. Zhu, T. J. Bethea, and V. Krishna. Extracting relevant named entities for automated expense reimbursement. In *KDD*, 2007.

[30] V. Romero and J. A. Sánchez. Category-based language models for handwriting recognition of marriage license books. In *Document Analysis and Recognition (ICDAR), 2013 12th International Conference on*. IEEE, 2013.

[31] J. I. Toledo, S. Sudholt, A. Fornés, J. Cucurull, G. A. Fink, and J. Lladós. Handwritten word image categorization with convolutional neural networks and spatial pyramid pooling. In *Joint IAPR International Workshops on Statistical Techniques in Pattern Recognition (SPR) and Structural and Syntactic Pattern Recognition (SSPR)*. Springer, 2016.

[32] A. Prasad, H. Déjean, J.-L. Meunier, M. Weidemann, J. Michael, and G. Leifert. Bench-marking information extraction in semi-structured historical handwritten records. *arXiv preprint arXiv:1807.06270*, 2018.

[33] M. Carbonell, M. Villegas, A. Fornés, and J. Lladós. Joint recognition of handwritten text and named entities with a neural end-to-end model. *arXiv preprint arXiv:1803.06252*, 2018.

[34] C. Adak, B. B. Chaudhuri, and M. Blumenstein. Named entity recognition from unstructured handwritten document images. In *Document Analysis Systems (DAS), 2016 12th IAPR Workshop on*. IEEE, 2016.

[35] H. Liu, J. Feng, M. Qi, J. Jiang, and S. Yan. End-to-end comparative attention networks for person re-identification. *arXiv preprint arXiv:1606.04404*, 2016.

[36] P. Krishnan, K. Dutta, and C. Jawahar. Deep feature embedding for accurate recognition and retrieval of handwritten text. In *Frontiers in Handwriting Recognition (ICFHR), 2016 15th International Conference on*. IEEE, 2016.

[37] A. Krizhevsky, I. Sutskever, and G. E. Hinton. Imagenet classification with deep convolutional neural networks. In *Advances in Neural Information Processing Systems*, 2012.

[38] K. He, X. Zhang, S. Ren, and J. Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition*, 2016.

[39] J. Lafferty, A. McCallum, and F. C. Pereira. Conditional random fields: Probabilistic models for segmenting and labeling sequence data. 2001.

[40] E. F. Tjong Kim Sang and S. Buchholz. Introduction to the conll-2000 shared task: Chunking. In *Proceedings of the 2nd workshop on Learning language in logic and the 4th conference on Computational natural language learning-Volume 7*. Association for Computational Linguistics, 2000.

[41] U.-V. Marti and H. Bunke. The iam-database: an english sentence database for offline handwriting recognition. *International Journal on Document Analysis and Recognition*, 2002.

[42] T. M. Rath and R. Manmatha. Word spotting for historical documents. *International Journal of Document Analysis and Recognition (IJDAR)*, 2007.

[43] A. Poznanski and L. Wolf. Cnn-n-gram for handwriting word recognition. In *CVPR*, 2016.

[44] S. N. Srihari, R. K. Srihari, P. Babu, and H. Srinivasan. On the automatic scoring of handwritten essays. In *IJCAI*, 2007.

[45] E. A. Kozak, R. S. Dittus, W. R. Smith, J. F. Fitzgerald, and C. D. Langfeld. Deciphering the physician note. *Journal of general internal medicine*, 1994.

[46] M. Rusiñol, D. Aldavert, R. Toledo, and J. Lladós. Efficient segmentation-free keyword spotting in historical document collections. *PR*, 2015.

[47] H. Wei, H. Zhang, and G. Gao. Word image representation based on visual embeddings and spatial constraints for keyword spotting on historical documents. In *2018 24th International Conference on Pattern Recognition (ICPR)*. IEEE, 2018.

[48] P. Krishnan and C. Jawahar. Hwnet v2: An efficient word image representation for handwritten documents. *arXiv preprint arXiv:1802.06194*, 2018.

[49] C. D. Manning, M. Surdeanu, J. Bauer, J. R. Finkel, S. Bethard, and D. McClosky. The stanford corenlp natural language processing toolkit. In *ACL (System Demonstrations)*, 2014.

[50] A. Sharif Razavian, H. Azizpour, J. Sullivan, and S. Carlsson. Cnn features off-the-shelf: an astounding baseline for recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, 2014.

[51] A. Sordoni, Y. Bengio, and J.-Y. Nie. Learning concept embeddings for query expansion by quantum entropy minimization. In *AAAI*, 2014.

[52] D. Metzler, S. Dumais, and C. Meek. Similarity measures for short segments of text. In *European Conference on Information Retrieval*. Springer Berlin Heidelberg, 2007.

[53] D. M. Blei, A. Y. Ng, and M. I. Jordan. Latent dirichlet allocation. *Journal of machine Learning research*, 3, 2003.

[54] A. K. McCallum. Mallet: A machine learning for language toolkit. http://mallet.cs.umass.edu, 2002.

[55] D. Bär, C. Biemann, I. Gurevych, and T. Zesch. Ukp: Computing semantic textual similarity by combining multiple content similarity measures. Association for Computational Linguistics, 2012.

[56] V. Rowtula, P. Krishnan, and C. Jawahar. Pos tagging and named entity recognition on handwritten documents. In *Proceedings of the 15th International Conference on Natural Language Processing*, 2018.

[57] Z. Huang, W. Xu, and K. Yu. Bidirectional lstm-crf models for sequence tagging. *arXiv preprint arXiv:1508.01991*, 2015.

[58] M. O. Dzikovska, R. D. Nielsen, and C. Leacock. The joint student response analysis and recognizing textual entailment challenge: making sense of student responses in educational applications. *Language Resources and Evaluation*, 2016.