Stroke Recognition from Characters

Interim Report of B.Tech IT 5^{th} Semester Mini-Project

Supervised By
Prof. Sudip Sanyal

Group Members

Kaustubh Shamshery
Pratik Mangalore
Prateek Agarwal
Sulyab Thottungal

Candidate's Declaration

We hereby declare that the project report titled "Stroke Recognition from Characters" submitted by us to Indian Institute Of Information Technology, Allahabad, in fulfillment of the requirement for the reward of a B.Tech in Information Technology, is a record of bonafide project work, under the supervision of Prof. Sudip Sanyal

Group Members

Kaustubh Shamshery Pratik Mangalore

Prateek Agarwal Sulyab Thottungal

Supervisor's Certificate

This is to certify that the project report entitled Stroke Recognition from Characters submitted to Department of Information Technology, Indian Institute of Information Technology, Allahabad in partial fulfillment of the 5^{th} semester Mini-Project work, is a record of
bonafide work carried out by :

1. Kaustubh Shamshery – IIT2014147
2. Pratik Mangalore – IIT2014112
3. Prateek Agarwal – IIT2014068
4. Sulyab Thottungal – IIT2014125

under my supervision and guidance.
No part of this report has been submitted elsewhere for any other purpose.

Prof. Sudip Sanyal
Dean (Research and Development)
Department of Information Technology
Indian Institute of Information Technology
Allahabad 211 012, India

Contents

1 Abstract

Background : Probabilistic Latent Semantic Analyis is a statistical technique used for the analysis of co-occurence data. In effect, one can derive a low-dimensional representation of the observed variables in terms of their affinity to certain hidden variables. We apply the above technique on a dataset consisting of images of characters in order to derive strokes which can generate the complete alphabet. The strokes here are the hidden variables which are a low-dimensional representation of the observed variables - pixels, in this case.

Purpose of Research : The human brain learns by recognizing patterns in the data that it is fed. Similarly, it is seen that we recognize the characters in an alphabet based on the different kinds of strokes they have. Here, a stroke is a unique formation of pixels, which co-occur regularly in the characters and do not belong to other strokes. We are using the PLSA technique on the character images so as to find out these strokes in a manner not very unlike how we humans learn.

Methods Used : We are primarly using the following methods to generate the strokes :-
1) Probabilistic Latent Semantic Analysis
2) Expectation Maximization Algorithm (As an internal step in PLSA)

Current Result : By applying the PLSA technique (using EM) on the dataset, we have been able to obtain strokes which can be combined in order to recreate the characters in a given alphabet.

Conclusion : We conclude that the PLSA algorithm which until now is usually used in the domain of NLP can be modified so that it can be applied to images with greater levels of accuracy.

2 Introduction

2.1 Overview

Recognition of printed characters and handwriting is a subfield of prime importance in computer vision and artificial intelligence domains. Digitalization of information for safe storage and efficient manipulation has become a necessity in this era of information explosion.

Character recognition has been implemented very efficiently using many techniques in the past few decades. However, in this project, we explore a new method - identifying the strokes required to generate an entire alphabet. Here, strokes mean the basic curves which add up to produce whole letters.

2.2 Motivation

Identification of strokes is important as it gives and takes from how the human brain recognizes objects. The human brain uses information of orientation and spatial relations to identify objects.

We propound the idea that the identification of strokes using PLSA on the characters' images is analogous to the manner in which our brain learns and recollects information. Using this method to identify characters might, in turn, provide more insight into the intricate ways of brain functioning.

2.3 Scope

The latent variables generated in this algorithm are the strokes present in the alphabet itself, and they are learnt by the algorithm in an unsupervised fashion. This makes the algorithm script-independent. Also, the EM algorithm converges faster than the backpropagation algorithm which is used in neural nets, which is a popular alternative for character recognition.

2.4 Problem Definitions and Objectives

Stroke Definition :-
We define a stroke as a unique co-occurence of pixels which exhibit a specific spatial orientation.

Objective :-
Given a dataset consisting of grayscale images of characters in a script, the strokes that completely generate the script are to be identified using unsupervised learning approach. The probability that a particular pixel of a particular character is black, can be expressed

using the probability distribution of strokes as

$$P\left(x_i|c_j\right) = \sum_{k \in S} P\left(s_k|c_j\right) P\left(x_i|s_k\right) \tag{1}$$

where x_i is the state of the i^{th} pixel (black/white), c_j is the j^{th} letter in the script, s_k is the k^{th} stroke that has been identified, and S is the set of all strokes.

The above expression contains latent (hidden) variables in its RHS, and hence cannot be determined directly. Hence, we adopted the PLSA (Probabilistic latent semantic analysis) model, which is a statistical technique for automated document indexing. In PLSA, the probability of each variable is modelled as expressions involving conditionally independent multinomial distributions.

2.5 Organization of the Project

For effecient collaboration, we used Google Drive and a private GitHub repository.

Each one of us had our own working dataset on which ran the algorithm separately, each time changing the definitons or number of sample/stroke.

3 Literature Survey

3.1 Background and Related Work

Sr.No	Author/Website/Book	Paper Title	Referenced For
1	Thomas Hoffman	"Probabilistic latent semantic analysis," in Proceedings of the fifteenth conference on uncertainty in artificial intelligence, 1999, pp. 289-296.	Theory related to the PLSA algorithm and how topic categorization relates to dividing characters into strokes.
2	Thomas Hoffman	"Unsupervised learning by probabilistic latent semantic analysis," Machine learning, vol. 42, iss. 1-2, pp. 177-196, 2001	How the EM algorithm is used in PLSA in order to maximize the likelihood of presence of stroke in a character such that a pixel exists in it.
3	DataEra.org	Notes on EM and pLSA"	For referring the mathematical steps used in the EM algorithm.

3.2 Concluding Remarks on the basis of Literature Survey

After completing the research survey it was found that, the PLSA algorithm could in fact be used on images with definite success. We know this because the definitions in the case when PLSA is applied on document and words are similar to the definitions when PLSA is applied on images.

4 Plan Of Work

4.1 Proposed Approach

We divided our work into the following parts :-

4.1.1 Preprocessing

- At the start all the sample images were of different sizes. Therefore, we decided to resize them into a square image of size 40. Not only would this make the image uniform, but it would also reduce the running time of the algorithm.
- We also generated the Term Document Matrix [1], which will be used by the PLSA algorithm. The 2517 rows in the Term Document matrix represent the images in the dataset whereas the 1600 columns represent the grayscale value of each pixel.

4.1.2 Definitions

Firstly, lets define a few entities.

- Let N be the number of image samples (this is analogous to number of documents) which we have currently taken to be 2517.
- Let M be the number of pixels in our images (this is analogous to number of words in our vocabulary) which is clearly 1600 (40x40).
- w_j refers to the j^{th} pixel (i.e, 0 to 1599) being black, and d_i refers to the i^{th} image (i.e, 1 to 2517)
- The below equation gives the log-likelihood function which we want to maximize in the general EM algorithm

$$L(\theta) = \log P(y|\theta) = \log \sum_{z} P(y, z|\theta) \tag{2}$$

  The following equation [2] shows the log-likelihood function obtained from the PLSA model

$$logP(d, w) = \sum_{i=1}^{N} \sum_{j=0}^{M-1} n(d_i, w_j) logP(d_i, w_j) \tag{3}$$

- In the log-likelihood function (3), we find a log structure with latent variable that is very similar to the log-likelihood function (2) of a general EM problem. Hence, we can apply EM algorithm to the PLSA model to maximize the log-likelihood.

4.1.3 Applying EM algorithm on the PLSA model

Few more definitions for the PLSA model [1] :-

- $P(z_k|d_i, w_j)$ is the posterior probability that k^{th} stroke exists given that i^{th} image is considered and j^{th} pixel is black.
- $P(w_j|z_k)$ is the probability that the j^{th} pixel is black and exists in the k^{th} stroke.
- $P(z_k|d_i)$ is the probability that the k^{th} stroke exists in i^{th} image.

The Expectation-Maximization [3] algorithm, applied to PLSA model is as follows :

Initialization : Initialize the parameters $P(z|d)$ and $P(w|z)$ randomly

Repeat until convergence :

- E-STEP :-

$$P\left(z_k|d_i, w_j\right) = \frac{P\left(w_j|z_k\right)P\left(z_k|d_i\right)}{\sum_{l=1}^{K} P\left(w_j|z_l\right)P\left(z_l|d_i\right)}$$

- M-STEP :-

$$P\left(w_j|z_k\right) = \frac{\sum_{i=1}^{N} n\left(d_i, w_j\right)P\left(z_k|d_i, w_j\right)}{\sum_{m=1}^{M}\sum_{i=1}^{N} n\left(d_i|w_m\right)P\left(z_k|d_i, w_m\right)}$$

$$P\left(z_k|d_i\right) = \frac{\sum_{i=1}^{N} n\left(d_i, w_j\right)P\left(z_k|d_i, w_j\right)}{n(d_i)}$$

4.2 Dataset Description

The original dataset contained two groups of characters which represent the consonants and the vowels of the Tibetan script. The first group had 211 characters, each having 20-50 samples. The original dataset contained images of non-uniform dimensions. We first resized the images to 40x40 for uniform sizing and reducing the running time of the program.

4.3 Software Requirements

Hardware

- Python (including numpy, matplotlib, os, PIL, datetime packages) [4]
- Git and GitHub

4.4 Activity Time chart

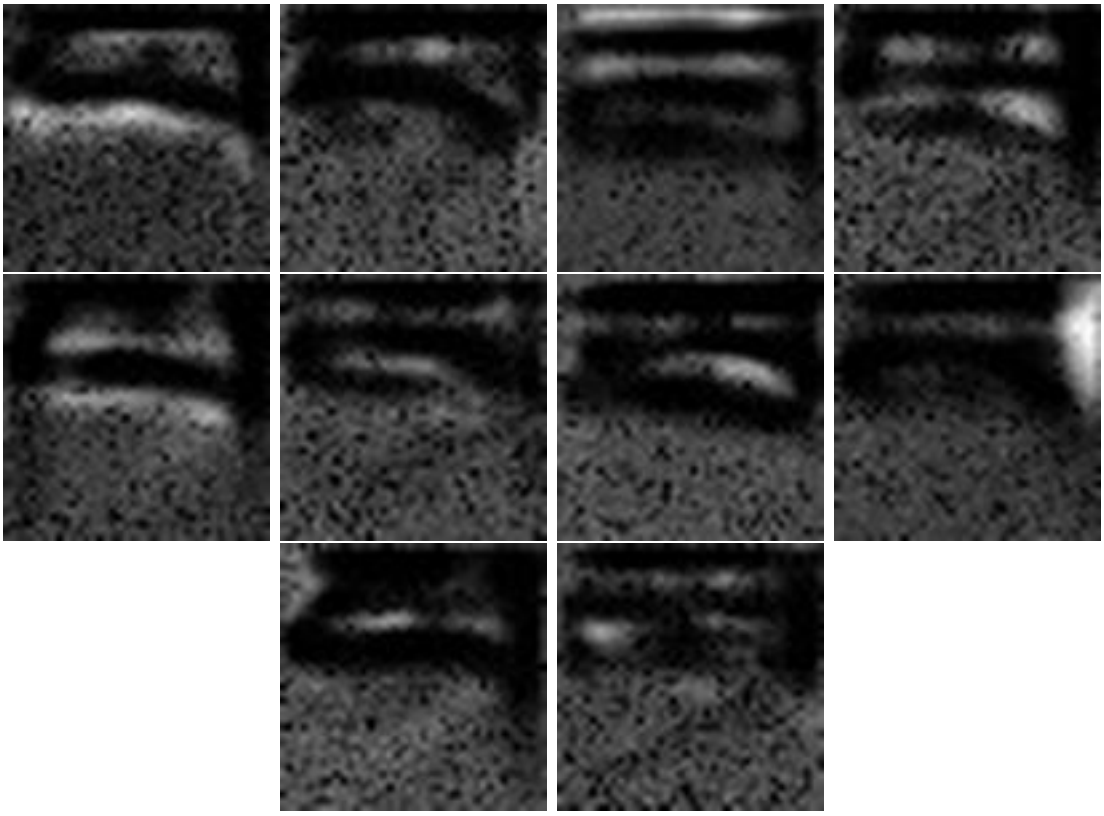
The task was divided as follows:-

Task	Expected time
Literature Survey	2 weeks
Problem Identification	1 week
Project Implementation	1 week

5 Results and Analysis

5.1 Results :

- We ran the EM algorithm on our PLSA model by passing the number of strokes as a parameter. Different stroke patterns were obtained for different input.
- We observed that the stroke patterns generated depended on number of strokes, size of the dataset used and the number of iterations.
- $P(w_j|z_k)$ and $P(z_k|d_i)$ were observed to converge after approximately 100 iterations of the EM algorithm.



5.2 Analysis :

- Depending on the size of dataset we used to train the algorithm, the time taken for generating the strokes ranged from 1-5 hrs. The strokes generated were much more accurate and clear when the PLSA model ran on a largeer dataset.
- The time complexity of our program is $O(n^4)$ which equates to about 10^{10} instructions of python code to execute

6 Conclusion and Future Scope

- Decompose the 'strokes' obtained into simpler, basic strokes without changing the unsupervised nature of the algorithm
- Optimize the algorithm and reduce its running time
- Introduce modifications to make sure EM does not get stuck at local maxima
- Devise a method to extract the minimum number of strokes required to completely and accurately generate a given script, while taking care of overfitting issues

References

[1] Thomas Hoffman. "Probabilistic Latent Semantic Analysis". In: *Proceedings of the fifteenth conference on uncertainty in artificial intelligence,* (1999), pp. 289–296.

[2] Thomas Hoffman. *Unsupervised Learning by Probabilistic Latent Semantic Analysis*. Vol. 42. Machine Learning. 2001.

[3] *Notes on EM and PLSA*. URL: `http://dataera.org/2014/04/notes-on-em-and-plsa/`.

[4] *Python*. URL: `https://www.python.org`.

7 Suggestions And Remarks :

Please write down your suggestions and remarks if any: