Scene Classification via pLSA

Sean Hoessmann
V00850544
Faculty of Engineering
University of Victoria
Victoria, Canada

Purvika Dutt
V00849852
Faculty of Engineering
University of Victoria
Victoria, Canada

Patrick More
V00840228
Faculty of Engineering
University of Victoria
Victoria, Canada

***Abstract*—The objective of this project is to give a correct classification for a given image. A bag of words is used to apply tags to the features within an image before using probabilistic Latent Semantic Analysis to improve classification. k-nearest neighbor is used as the classifier. The algorithm and approach is based on the implementation by Bosch. et al. [1] in their report titled "Scene Classification via pLSA". The results are then evaluated against targeted accuracy values as well as the original reports' results. The dataset used is a subset taken from the Places dataset [2].**

I. Introduction

II. Literary Review

Scene classification is a difficult task, and has been implemented in various fashions using different algorithms. These implementations use either supervised or semi-supervised algorithms for the scene classification. Bosch et al.'s algorithm [1] uses both unsupervised pLSA and supervised k-nearest neighbor algorithms in order to improve upon existing algorithms, and gain a higher level of accuracy when classifying scenes.

Fei Fei and Perona [3] use Latent Dirichlet Analysis in their report titled "" which is a semi-supervised algorithm. In their article, they concluded that their model is incomplete due to lackluster performance in indoor scenes. Their performance also suffered when using unsupervised learning, requiring human annotation of 6 properties for thousands of scenes in order for their algorithm to function ideally.

Other implementations use a fully supervised approach, which requires some form of manual classification of the images. Both Vogel and Schiele's [4] implementation as well as Oliva and Torralba's [5] implementation use supervised algorithms. Vogel and Schiele's immplementation suffered from low accuracy for some types of scenes, with rivers/lakes being largely misclassified, and in order to improve accuracy for those classifications would heavily impact the other classifications making such changes inoptimal.

Bosch et. al.'s algorithm improved upon those existing scene classification methods and used the same datasets in order to compare directly. For the implementation done in this report, we instead use a new dataset so it can no longer directly compare results against previous algorithms. This report will instead be observing how the pLSA algorithm performs on a new, different dataset.

The algorithm in this paper attempts to classify the scene using the bag of words model with the pLSA algorithm. In essence this algorithm learns the conditional distribution of the document topics given the words. The topics are learned during this distribution fitting and the words are clustered using the k-means algorithm. We can calcluate a probababilty vector containing this conditional distribution for any image we wish to classify. The classification is performed on this vector of probabilities, called a Z vector, using the k-nearest neighbor algorithm.

This approach has a number of weaknesses, the largest being that this approach doesn't capture any data about the spatial co-occurence of visual words. This causes issues when attempting to classify reflective surfaces, e.g. the clouds in the reflection on water may be classified as sky. In addition more advanced clustering algorithms and classification algorithms than kmeans and k-nearest neighbor could potentially improve performance. Although this would depend greatly on how the data is typically distributed in the Z vectors of image data. In addition this algorithm requires large amounts of memory to train the model on large datasets, as it considers the whole dataset at once, once a trained the requirements go down drastically.

III. Proposed Approach

The proposed approach of the source paper uses a fairly common algorithm in the natural language processing realm, probabilistic latent semantic analysis, in a fairly novel way to classify images. To keep terminology consistent with the NLP terminology images will be referred to as documents and visual words will be referred to as words. This algorithm uses the bag of words model, where each word is a feature vector extracted from the image, which is then clustered into a category of visual word. The pLSA algorithm models unknown latent variables, which will be referred to as topics, within each document and constructs a conditional probability distribution of the topics given the words. This is done by using an Expectation Maximization(EM) algorithm over all the words in the training data. The Expectation portion of

the algorithm captures the distribution of the topics over the documents and words of the training set

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

In the Maximization step the expected log likilhood of the current values are calculated using

Then the conditional distributions are updated using the co-occurence table found in the inital feature clustering.

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

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

The E-step and M-step are alternated between until one of two terminating conditions is met. The first condition is that the difference in log-liklihood is suffeciently close to zero and the second is if the maximum number of iterations has been exceeded. The estimated conditional distribution of the words given the topics is kept and is used in the classification portion of the algorithm.

Classifying an image follows the same process as the training with one key difference. When classifying an image the distribution $P(w_j|z_k)$ is not updated, instead using the distribution found in the training phase of the algorithm. When the EM algorithm is finished running on the image being classified, since their is only one choice of documet, the only fitted distribution is the topics given the document which is simply the vector of topic probabilities for the document being classified.

$$Z_k = P(z_k|d_i)$$

The Z vector of an document is a vector of this distribution over all topics. The Z vector of a document is a much lower dimensional space than either the document or the visual words, which allows more powerful classifiers to be used at a lower cost. To classify the image a K-nearest neigbors algorithm is run using the Z vector of the image and the Z vectors of the training dataset.

A. Flow-Chart

Shown below is the diagram for our proposed implementation approach.

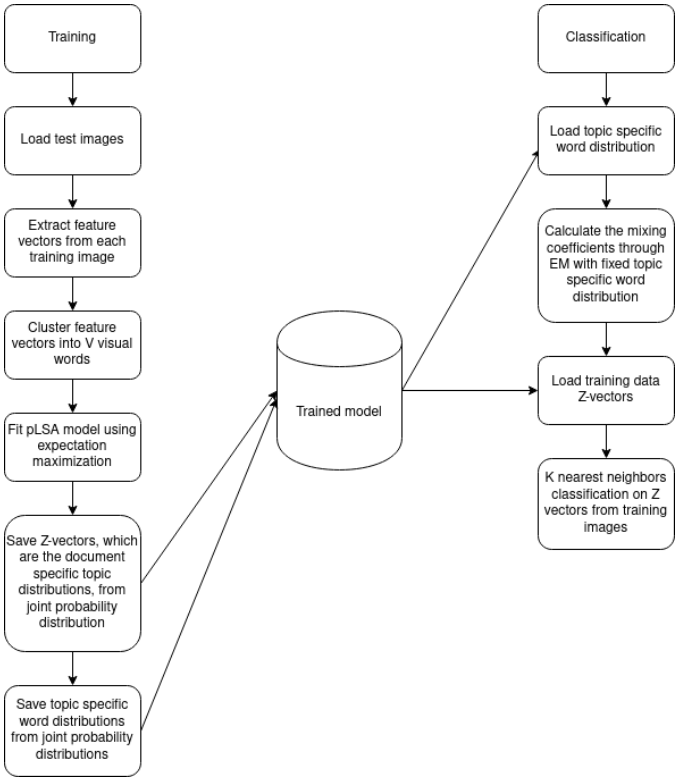

Fig. 1. Flow-Chart.

B. Validation

The algorithm can be split into three sections: feature extraction, expectation maximization, and k-nearest neighbor. The feature extraction portion of the paper has two methods for extracting features: patch and SIFT. The patch extraction is simply a square region of the image of length N, which is then flattened into a NxN length vector. The SIFT descriptor is a sift descriptor calculated on keypoints over a square region. Both of these methods are calculated over a grid with regular spacing between the keypoints. Both of these can be verified to a high degree of accuracy by visually inspecting the output of the patches and comparing to other implementations for the SIFT descriptors.

It will be hard to verify the expectation maximization past some more trivial low-dimensionality examples. However since the equations for EM are made up of structured multiplications and summations, as seen below, if our implementation is structured properly we can be confident in a correct solution. The two steps for expectation maximization can be seen below where z is a topic, d is a document, w is a word, and n is the co-occurrence table.

Expectation:

$$P(z|d, w) = \frac{P(z)P(d|z)P(w|z)}{\sum_{z'} P(z')P(d|z')P(w|z')}$$

Maximization:

$$P(w|z) = \frac{\sum_d n(d,w)P(z|d,w)}{\sum_{d,w'} n(d,w')P(z|d,w')}$$

$$P(d|z) = \frac{wn(d,w)P(z|d,w)}{\sum_{d',w} n(d',w)P(z|d',w)}$$

$$P(z) = \frac{1}{R}\sum_{d,w} n(d,w)P(z|d,w)$$

$$R = \sum_{d,w} n(d,w)$$

To implement this portion we'll begin with the standard EM algorithm, before modifying to use the tempered EM algorithm. As the difference between the two algorithms is small, as can be seen below, this will allow us to build out the algorithm iteratively. The symbols are the same as above and is used to temper the results, it starts at 1 and slowly decreases towards zero throughout the iterations.

$$P_\beta(z|d,w) = \frac{P(z)[P(d|z)P(w|z)]^\beta}{\sum_{z'} P(z')[P(d|z')P(w|z')]^\beta}$$

The k nearest neighbor is an easy algorithm to implement naively, and the naive version can be fairly quickly verified. This can be implemented by keeping a sorted list of the distance to the nearest neighbors seen so far and tracking the corresponding classes. This is lower on the priority list and if need be we will use a pre-made implementation as it is the least interesting part of the algorithm.

C. Assumptions

Our chosen algorithm makes a number of assumptions. The first assumption is that the training data is representative of any further data it classifies. The second is that the words and documents are conditionally independent when conditioned on the class of the image. This assumption comes directly from the fact that pLSA is a generative model. If the first assumption is wrong we can expect to see a reduced accuracy across all classes, as the probability distributions calculated in the training stage will be incorrect. If the second assumption is incorrect we can expect to see a large decrease in the classifiers accuracy as this assumption is core to the pLSA model.

IV. Evaluation & Datasets

V. Conclusion

References

[1] A. Bosch, A. Zisserman, and X. Muñoz, "Scene Classification via pLSA", Computer Vision – ECCV 2006, pp. 517-530 Pattern Analysis and Machine Intelligence, 2017

[2] B. Zhou, A. Lapedriza, A. Khosla, A. Oliva, and A. Torralba, "Places: A10 million Image Database for Scene Recognition", IEEE Transactions on Pattern Analysis and Machine Intelligence, 2017

[3] L. Fei-Fei, P. Perona, "A bayesian hierarchical model for learning natural scene categories", CVPR, Washington, DC, USA, (2005) 524–531

[4] J. Vogel, B. Schiele, "Natural scene retrieval based on a semantic modeling step", CIVR, Dublin, Ireland (2004)

[5] A. Oliva, A. Torralba, "Modeling the shape of the scene: a holistic representation of the spatial envelope", IJCV (42) 145–175