Genre-based Image Classification Using Ensemble Learning for Online Flyers

Payam Pourashraf[a], Noriko Tomuro[b], Emilia Apostolova[c]
[a,b]DePaul University, 243 S. Wabash Ave, Chicago, IL 60604 USA
[c]BrokerSavant Inc., 2506 N. Clark St. Chicago, IL 60614 USA
ppourash@cdm.depaul.edu, tomuro@cs.depaul.edu, emilia@brokersavant.com

ABSTRACT

This paper presents an image classification model developed to classify images embedded in commercial real estate flyers. It is a component in a larger, multimodal system which uses texts as well as images in the flyers to automatically classify them by the property types. The role of the image classifier in the system is to provide the *genres* of the embedded images (map, schematic drawing, aerial photo, etc.), which to be combined with the texts in the flyer to do the overall classification. In this work, we used an ensemble learning approach and developed a model where the outputs of an ensemble of support vector machines (SVMs) are combined by a k-nearest neighbor (KNN) classifier. In this model, the classifiers in the ensemble are *strong* classifiers, each of which is trained to predict a given/assigned genre. Not only is our model intuitive by taking advantage of the mutual distinctness of the image genres, it is also scalable. We tested the model using over 3000 images extracted from online real estate flyers. The result showed that our model outperformed the baseline classifiers by a large margin.

Keywords: image classification, ensemble learning, image genre, support vector machine, flyers, embedded images

1. INTRODUCTION

In recent years, as the speed and bandwidth with which people can access internet has increased enormously, online information available on the internet has become overwhelmingly multimedia. Almost every post on a social network site includes snap photos or video clips, while most web pages nowadays are rich with graphics and embedded multimedia components. Commercial flyers posted on the internet are an example of such online multimedia content (a.k.a "*infographics*"). Typically a flyer contains textual descriptions such as the title/name of the subject matter and the relevant information, and some images such as pictures and logos/icons. The two modalities complement each other in conveying information -- texts provide relevant information explicitly by words, while images provide information (additional as well as relevant) implicitly through visual representations. The use of images is extremely important for commercial flyers in increasing the effectiveness of marketing.

In this paper, we present preliminary results of our work on classifying images embedded in commercial real estate flyers. Figure 1 shows an example (2-page) flyer for an industrial property. Brokers of commercial real estate have a collection of properties which they sell, and for each property they create a flyer, usually in pdf and/or html email or web page, with all relevant listing information to market the property. Brokers these days also collect information on other available properties from other brokers or public flyers, and build a searchable database to attract clients. However, getting the relevant information out of a flyer and manually entering data in a database is a tedious task and error-prone. A better approach is to automatically do the extraction and index the flyers.

With real estate flyers, most key information on the property is usually in written in text, for example the square footage, the price/rate and the property type (e.g. retail, office, industrial, land, etc.). However, automatic extraction of such information is not as straight-forward as it may seem, mostly due to the free formed-ness of the flyers -- Since flyers do not have a fixed structure, it is difficult to identify relevant pieces of text with high accuracy.

In this paper, we describe our work on classifying images embedded in real estate flyers by *image genre* (map, schematic drawing, aerial photo, etc.). It is a part of a larger project which aims to develop a high-performance multimodal system which extracts information from real estate flyers by using both texts and images. In this paper, we focus on the image part, and describe our methods from image pre-processing, feature extraction, to classification. The role of the image classifier in the system is to provide the genres of the embedded images, which to be combined with the texts in the flyer to do the overall classification. Our work is unique in that, not only is it a part of an application which has a practical import, we also developed a new image classifier based on ensemble learning which is intuitive as well as *scalable*. The results showed the new classifier produced significantly improved performance over standard baseline classifiers as well.

Figure 1. A commercial real estate flyer of an industrial property (© Lee & Associates).

2. RELATED WORK

While previous works on image classification have predominantly used image content (e.g. sunset, beach, flowers) to classify images, there are only a few which classified based on image genres. [1] presents a system which categorizes images into three genres (art, photo, cartoon). They used the standard MPEG-7 visual descriptors as the image features and built a classification system using Neural Networks. An interesting work would be [2] which classified digital images of paintings by artistic genre (e.g. Impressionism, Abstract Expressionism).

Ensemble learning, on the other hand, has been used in many previous image classification works. Also a number of them used SVM in the ensemble. Recent works include [3] and [4]. However, SVM is a binary classifier, thus has to be adapted in some way to work with multiclass problems. One of the seminal works on that topic (although not for image classification) was by [5], and among the recent works [6] conducted extensive experiments to investigate various ensemble schemes to adapt SVM to multiclass problems in image classification.

3. THE IMAGE DATASET

3.1 Image Extraction from Flyers

In this work, we created our image dataset by extracting images from the commercial real estate flyer dataset used in [7]. The flyer dataset contained 800 files in the pdf format. To extract images, we first converted each pdf file to the html format – through that process we obtained the embedded images as png files as a byproduct, and then cropped the ones which had multiple images in one file into individual images.[a] This process yielded a total of 6758 images. However some of them were 'noisy' images – for example, fragments of a larger image which were produced by cropping error, or strips of color borders, company logos or QR codes which were irrelevant for our purpose. To filter them, we wrote a small program and automatically removed them from the set. This process left us a total of 3416 images, and that constituted our image dataset. Finally we manually tagged each image as one of the five *genre* categories: Aerial Photo, Map, Schematic Drawing, Inside Building and Outside Building. Note that the genre classes were not predetermined -- we selected ones based on what we felt represented the *types* of the images in our dataset most naturally. The distribution of the number of images (and the proportion in the dataset) was as follows: Aerial photo (362, 10.6%), Map (834, 24.4%), Schematic drawing (556, 16.3%), Inside building (690, 20.2%), Outside building (974, 28.5%).

3.2 Feature Extraction from Images

After tagging the images with genres, we extracted some image features, which to be fed into the classification algorithms. In particular, for each image we calculated (1) Autocorrelogram [8], (2) Tamura [9], (3) Local Binary Patterns (LBP) [10] and (4) Histogram of Oriented Gradients (HOG) [11, 12]. For (1) Autocorrelogram, we first

[a] We used pdftohtml (http://sourceforge.net/projects/pdftohtml/) to convert pdfs to html, and Gimp (http://www.gimp.org/) to crop individual images.

quantized the R,G,B color channels to 4 levels (thus totaling 64 colors), then calculated the correlograms for distance 1 and 3, thereby obtaining a total of 128 (=2*64) features. For (2) Tamura, we extracted the first three features (coarseness, contrast, directionality; most important ones in the total six features [13]). We chose Tamura features (over Haralick) because they correlate well with human visual perception [9]. For (3) LBP, we used the basic LBP(8, 1), which considers 8 neighbors with distance one. That yielded a histogram with 256 bins. Then we quantized the bins to 32, to obtain 32 features. For (4) HOG, we used 9 rectangular cells, each of which quantized to 9 bins, and obtained a total of 81 features. Finally we also computed the number of lines (by using Hough Transform [14]) and the number of points with high cornerness (by using Harris corner detection [15]) as additional features. We included those features because we thought they would give distinguishing values for particular genres (such as inside/outside buildings over maps and drawings). By putting together these features, we obtained the final feature vector of length 246 (1x246) for each image.

3.3 Preliminary Experiments

After extracting image features, we conducted a brief preliminary experiment to obtain a rough idea on the general complexity of the data, that is, the mutual exclusiveness/distinctness of the genres. To that end, we chose three algorithms (Naïve Bayes, K-nearest neighbor (KNN) and Decision Tree) and classified the data. We chose those algorithms because, among various classification algorithms, they naturally permit multiclass problems (as versus algorithms which fundamentally assume binary classification). In that sense, those algorithms also serve as a baseline to which we will be able to compare as we develop our model. Table 1 below shows the classification results. Note that the accuracies were obtained by randomly partitioning the dataset (consisting of 3416 instances) into 66% training (2277 instances) and 34% testing (1139 instances),[b] then building a model using the training set and testing the model with the test set; and for each algorithm we repeated the process three times and computed the average of the three runs.

Table 1. Preliminary multiclass classification results (baseline)

Algorithm	Naïve Bayes	KNN (K=5)	Decision Tree
Accuracy (%)	65.35	72.14	76.32

Contrary to our expectation, the accuracies turned out rather low (in the mid 60-70% range) – while we had anticipated relatively high accuracy because each genre looked fairly unique and distinct from others. For example, almost all images in the Drawing category had very little color variation (typically black/white), while most Maps followed the same color schema (e.g. orange for highways, red/green/white for highway number signs), but Aerial photos were predominantly green. There were seemingly quite distinct characteristics in the texture as well. So we ran another preliminary experiment focusing on the distinctness of individual genres/categories. For each category, we ran the "*one vs. others"* binary classification using the same three training/testing partitions, and computed the average accuracy. Note that we used Decision Tree for this experiment. Table 2 below shows the results.

Table 2. Binary classification accuracy by image genre

Genre/Category	Aerial Photo	Map	Schematic Drawing	Inside Building	Outside Building
Accuracy (%)	95.14	93.07	93.88	88.53	84.43

As you see, the accuracies for individual genres are generally quite high (in the mid 80 to 90% range) – supporting our original intuition. This means the genres in our data are indeed distinct individually, but taken together as a multiclass problem, the hypothesis space is rather complex.

4. ENSEMBLE CLASSIFIER

In this work, we developed a classification model based on ensemble learning [16]. In Machine Learning, ensemble learning aims to obtain an accurate classifier by combining multiple classifiers. Rather than building a single strong classifier that covers the entire hypothesis space, the idea is to use an ensemble of weak classifiers, each of which covers a subspace of the hypothesis space, and combine them in some way to induce a strong classifier. Classifiers in an ensemble (Tier-1 classifiers) receive the input directly, and the combining meta-level classifier (Tier-2 classifier) receives the outputs of the Tier-1 classifiers and produces the final output. Figure 2 shows a diagram of a general ensemble model.

[b] We also used stratified partitioning: the class distribution of the target attribute in the original dataset was preserved in all subsets.

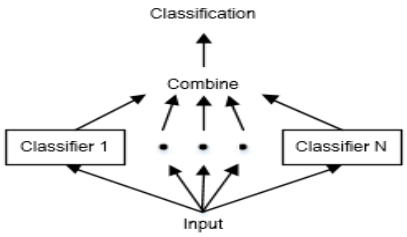

Figure 2. General Ensemble Model

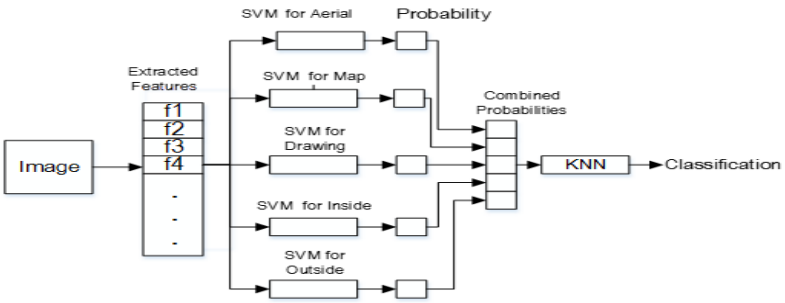

Figure 3. Our Ensemble Model

There have been several ensemble algorithms developed, including bagging, boosting and stacking. Our model is a variation of stacking, and resembles most closely to an algorithm called *mixture of experts* [17][18]. In this algorithm, the Tier-1 classifiers are essentially 'experts' trained on different target classes, and the Tier-2 classifier is a 'gating network' that decides which expert to use [17].

Figure 3 shows a schematic diagram of our ensemble model. There are five classifiers in the Tier-1 level, each of which is a binary support vector machine (SVM), trained to make prediction on a single category from all other categories (*one vs. others*). Output of a Tier-1 classifier is a probability, and the outputs from the five SVM classifiers are concatenated into a vector. The next level Tier-2 classifier is a KNN classifier, which receives a probability vector from Tier-1 and outputs the final classification for the instance.

Compared to other ensemble methods, our model is unique in several ways. First, all Tier-1 classifiers are a SVM, which has been shown in many previous works to produce higher accuracy than other classification algorithms. In other words, our Tier-1 classifiers are *strong* classifiers (whereas most ensemble learning uses weak classifiers). Second, the Tier-2 meta classifier is a non-linear classifier (KNN in particular) instead of a usual linear function (such as average, maximum and majority [17]). The motivation behind this model was from the preliminary experiments described in the previous section – our genres are relatively distinct individually, but when together they form a complex hypothesis space. So we chose to use an ensemble of strong Tier-1 classifiers, with an expectation that each classifier would produce high accuracies. As for Tier-2, we chose to use a non-linear classifier primarily so that the "ties" in the output probability vector produced by the Tier-1 classifiers are resolved in a more complex way.

Note that the *one vs. others* scheme we used for the Tier-1 SVM classifiers is one of the strategies to adapt binary classification algorithms to multiclass problems, and contrasts with another scheme called *one vs. one.* We chose *one vs. others* because of its efficiency and scalability: for a K-class problem, the *one vs. others* scheme creates K classifiers (one for each class) whereas the *one vs. one* scheme creates ½ K*(K-1) classifiers to do pair-wise comparisons [5]. While the *one vs. one* scheme could form more complex decision surface, thus potentially produce more accurate predictions, it does not scale up for larger problems.

5. EXPERIMENTAL RESULTS

We evaluated our model by comparing with different configurations of Tier-1 and Tier-2 settings. In particular, for Tier-1 we compared weak (Decision Tree) vs. strong (SVM) classifiers; and for Tier-2 we compared linear (maximum) vs. non-linear (KNN) algorithms. We chose Decision Tree as a weak classifier only relative to SVM (which we chose as the strong classifier). Table 3 below shows the results. Note that each run of the experiment was conducted using the same partitioned subsets and in the same way as the preliminary baseline classifiers.

As shown in the table, for Tier-1 the use of strong classifiers produced higher accuracy (combined with either Tier-2 classifier: 73.60 vs. 90.95, and 76.03 vs. 90.75; the differences were statistically significant with the p-value < 0.01 for both cases). However for Tier-2, the results were inconclusive as to whether or not the non-linear algorithm performed better: for weak Tier-1 classifiers (73.60 vs. 76.03) the p-value was < 0.01, but for strong Tier-1 classifiers (90.95 vs. 90.75) the p-value was > 0.05. This means the higher complexity for theTier-2 classifier may not always bring better performance. In fact, the same result has also been reported in some previous works such as [6]. Our further investigation revealed that the reason was the same values outputted from multiple Tier-1 classifiers – from the perspective of Tier-2 the same input values are indistinguishable, therefore it is difficult to produce more accurate predictions than simple linear functions.

Table 3. Classification accuracies by various Tier-1/Tier-2 configurations (%)

Tier-1 \ Tier-2	Linear (maximum)	Non-linear (KNN, N=5)	p-value
Weak (Decision Tree)	73.60	76.03	< 0.01
Strong (SVM)	90.95	90.75	> 0.05
p-value	< 0.01	< 0.01	

Lastly, we must note that the accuracies by the strong Tier-1 ensembles were much higher than the baseline results (shown in Table 1): dramatic increases from the mid 60-70% to 90%. Also the difference was statistically significant (for all combinations of comparison).

6. CONCLUSION AND FUTURE WORK

In this paper, we presented our classification model for classifies images embedded in real estate flyers by their genres. Our model is an ensemble of strong SVM classifiers, and outperforms baseline classifiers by a large margin. Our model is also intuitive, reflecting the mutual distinctness of the genres, as well as scalable because the number of ensemble classifiers only grows linearly with respect to the number of target classes. For future work, we plan to experiment with deep learning to investigate the possible performance gain by the complex multi-level architecture.

REFERENCES

[1] Lee, J., Baik, S., Kim, K., Jung, C. and Kim, W., "IGC: an image genre classification system," Artificial Intelligence and Computational Intelligence. Lecture Notes in Computer Science Volume 7003, 360-367 (2011).

[2] Zujovic, J., Gandy, L., Friedman, S., Pardo, B. and Pappas, T., "Classifying paintings by artistic genre: An analysis of features & classifiers," in Proceedings of IEEE Int'l Workshop on Multimedia Signal Processing, 1-5 (2009).

[3] Malisiewicz, T., Gupta, A. and Efros, A., "Ensemble of exemplar-SVMs for object detection and beyond," in Proceedings of IEEE International Conference on Computer Vision (ICCV), 89-96 (2011).

[4] Varol, E., Gaonkar, B., Erus, G., Schultz, R. and Davatzikos, C., "Feature ranking based nested support vector machine ensemble for medical image classification," in the 9th IEEE Int'l Symposium on Biomedical Imaging (ISBI), 146–149 (2012).

[5] Hastie, T. and Tibshirani, R., "Classification by pairwise coupling," The Annals of Statistics 26 (2), 451-471 (1998).

[6] Goh, K., Chang, E. and Cheng, K., "SVM Binary Classifier Ensembles for Image Classification", in Proceedings of the Tenth International Conference on Information and Knowledge Management (CIKM '01), 395-402 (2001).

[7] Apostolova, E. and Tomuro, N., "Combining Visual and Textual Features for Information Extraction from Online Flyers," in Proceedings of Empirical Methods in Natural Language Processing (EMNLP-14), 1924-1929 (2014).

[8] Huang, J., Kumar, S. R., Mitra, M., Zhu, W. J. and Zabih, R., "Image Indexing Using Color Correlograms," IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR), 762-768 (1997).

[9] Tamura, H., Mori, S., and Yamawaki, T., "Textural features corresponding to visual perception," in IEEE Transactions on Systems, Man and Cybernetics, 8(6), 460-473 (1978).

[10] Ojala, T., Pietikainen, M. and Maenpaa, T., "Multiresolution gray-scale and rotation invariant texture classification with local binary patterns," in IEEE Trans. on Pattern Analysis and Machine Intelligence, 24(7), 971-987 (2002).

[11] Dalal, N. and Triggs, B., "Histograms of oriented gradients for human detection," IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR), 886-893 (2005).

[12] Junior, O. L., Delgado, D., Gonçalves, V. and Nunes, U., "Trainable classifier-fusion schemes: an application to pedestrian detection," in the 12th IEEE Intelligent Transportation Systems (ITSC), 1-6 (2009).

[13] Castelli, V. and Bergman, L., "Image databases," Jon Wiley & Sons (2002).

[14] Ballard, Dana H., "Generalizing the Hough transform to detect arbitrary shapes," Pattern Recognition, 13(2), 111-122 (1981).

[15] Harris, C. and Stephens, M., "A combined corner and edge detector," in Proceedings of 4th Alvey vision conference, 147-151 (1988).

[16] Schapire, R., "The Strength of Weak Learnability," Machine Learning, 5 (2), 197-227 (1990).

[17] Jacobs, R., Jordan, M., Nowlan. S. and Hinton, G., "Adaptive mixtures of local experts," Neural Computation, 3, 79-87 (1991).

[18] Nowlan, S. J. and Hinton, G. E., "Evaluation of Adaptive Mixtures of Competing Experts," Advances in Neural Information Processing Systems 3, Morgan Kaufmann: San Mateo, CA (1991).