CIS 520 Project Final Report

Woodpecker (Xiang Deng, Yiren Lu, Dongni Wang)

Fall 2015

For the final project, we developed a system for gender prediction (male/female) from the language of their tweets and the image they post with their twitter profile. We were given a training set of 4998 labeled training samples and a testing set of 4997 testing samples. Each sample has 5000 words features, 7 pre-extracted image features and 30000 raw RGB image pixel features.

In our system, we used seven classifiers on different feature sets and combined them using the stacking method. The six classifier are: a logistic regression model on words features, an ensemble model consists of 300 decision stump trees using LogitBoost on selected words and image features, a SVM model with intersection kernel on selected words and image features, a SVM model with intersection kernel on selected and normalized words and image features, an ANN model with 2 hidden layer each with 100 and 50 nodes, a SVM model with RBF kernel on PCA-ed HOG features on face-detected images, and a SVM model with RBF kernel on PCA-ed LBP features on face-detected images. For the stacking method, we took the raw outputs (probabilities) of the six basic models mentioned above trained with 80% of training samples and trained a logistic regression model using the other 20% of training sample. Our final full model achieved an overall accuracy of 92.42%. In order to meet the time and space constraint for the competition, we dropped the SVM model on PCA-ed LBP features and replaced the SVM model on PCA-ed HOG features with one bagging of logistic regression classifiers on raw HOG features. The submitted model for final competition achieved an accuracy of 91.04%.

In the following sections, we present the cross-validation accuracies of each methods we tried and discuss the rationale of our final model. We also provide some interesting visualization such as the most predictive words and the visualization of auto-encoder.

1 Methods

In this section, we report the results of multiple methods we tried for feature extraction, dimension reduction, and classification.

1.1 Data Preprocessing

1.1.1 Stemming and Stop Words

Conclusion: Did not work!

1.2 Feature Selection

To extract features from the raw word and image features, we experimented with multiple feature selection methods, including Information Gain, BNS

1.2.1 Information Gain Over Words Features

words visualization

1.3 Dimension Reduction

1.3.1 PCA

1.4 Classification on Words Features and Extracted Image Features

1.4.1 Linear Regression (L2 Regularization) + Sigmoid

1.4.2 Logistic Regression + PCA

1.4.3 Logistic Regression on Raw Features

1.4.4 Naive Bayes

1.4.5 Neural Network

1.4.6 LogitBoost

1.4.7 Kernel SVM

1.5 Classification on Image Features

1.5.1 Grey-scale Images

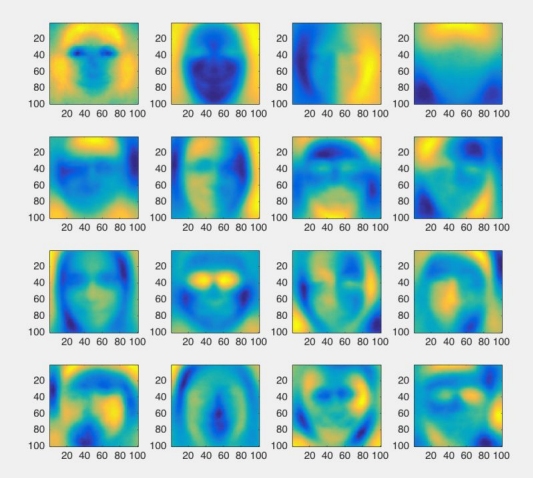
We first convert the profile images data into R, G, B and Grey-scale images.

1.5.2 Face Detection with Viola-Jones Algorithm

We extracted all the faces. We used matlab built-in Viola-Jones algorithm to do face detection on RGB images, which detected faces on 72% of the profile pictures. All the following classifications over images were based on the faces detected. Later when ensemble, we only used samples with detected faces. For those samples without faces in them, we just set the output of the classifier as 0 (the output of the classifier is either positive or negative for male or female).

1.5.3 Logistic Regression with Eigen Faces

After extracted faces, we did PCA over grey-scale images, the visualization of the top principal components are shown in figure below. We put top PCs into logistic regression classifier and it yielded 60+% accuracy.



1.5.4 SVM on PCA-ed HOG Features

1.5.5 SVM on PCA-ed LBP Features

1.5.6 Bagging Logistic Regression on PCA-ed HOG Features

1.6 Ensemble Methods

1.6.1 Stacking

We used stacking to ensemble our models to achieve better over all performance than any of the single models. First, we used 80% of the training data to train all the single models including logistic regression, neural network, logitboost with feature selection, kernel SVM with feature selection, kernel SVM with normalization, SVM over PCA-ed HOG, SVM over PCA-ed LBP. And then, use those models to generate predictions (scores) for the rest 20% data. Training a logistic regression model using the 20% scores with labels yields a ensemble classifier. Finally, we use all the training data to re-train all the single models. Along with the ensemble classifier, we got our final model. Our full model yielded 92.42% prediction accuracy on the leaderboard over test set. For the final submission, due to the space and time constraints, we dropped the LBP model and replaced the SVM over PCA-ed HOG with bagging logistic regression over raw HOG features. This final submission yielded 91.04% prediction accuracy on validation set, ranked 6th.

1.6.2 Normalization

When ensemble all the models, we notices that normalization actually works. For the images, we first trained logistic regression on raw gaussian pyramid HOG features (faces, eyes and nose), which yielded 82% accuracy on detected faces by itself. Later, we figured SVM works better by experiment (84%). However, when we ensemble the new model, it actually produced lower accuracy. We then found that the scores ranges produced by logistic regression and SVM were actually different. This may cause unbalanced weights across models. We then normalized all the scores with a sigmoid function with mean 0 and variance 2. This produced higher overall accuracy (92.3%). Tweaking the sigmoid function parameters for each model may produce a little bit higher accuracy but we didn't get enough time for that.

1.6.3 Cascaded Ensembling

2 Experiment Analysis

In this section, we analyze the results of our experiments of multiple methods for feature extraction, dimension reduction, and classification.

2.1 Feature Selection/Extraction

2.2 Dimension Reduction

2.3 Classification

The table of approaches and their associated 5-fold cross-validation classification accuracies are shown in the Table 1

3 Visualization

In this section, we include some interesting visualization obtained during the process of analyzing data, training, tuning, and testing our models.

Approach			
Feature	Dimension Reduction	Classifier	Cross-Validation Accuracy (%)
Words + Image features	PCA(500)	Ridge Regression + Sigmoid	$\approx 70\%$
Words + Image features	PCA(320)	Ridge Regression + Sigmoid	$\approx 79\%$
Words + Image features	PCA(2000)	Logistic Regression	$\approx 85\%$
Words	None	Logistic Regression	85.96%
Words	IG(1000)*	Logistic Regression	85.79%
Normalized-Words	None	Naive Bayes	72.25%
Words	None	multinomial Naive Bayes	63.59%
Words	IG(100)	multinomial Naive Bayes	77.95%
Words	None	Bernoulli Naive Bayes	79.59%
Words	IG(350)	Bernoulli Naive Bayes	81.75%
Words	IG(350)	Bernoulli Naive Bayes + EM	82.49%
Words	PCA(2000)	Artificial Neural Network	$\approx 86\%$
Words	None	K-Nearest Neighbor (L2)	$\approx 67\%$
Words	IG(76)	K-Nearest Neighbor (L2)	72.89%
Words	IG(84)	K-Nearest Neighbor (Minkowski)	71.43%
words	IG(95)	Random Forest	83.32%
words	IG(95)	Random Forest	83.32%
Words	None	K-means	$\approx 60\%$
Words	IG(1000)	Decision stumps + LogitBoost	89.11%
Face-detected Image RGB	PCA(100)	Random Forest	$\approx 69\%$
*IG represents Information Gain			

Table 1: Experimental results of single classifiers

	Approach	
Feature	Preprocessing	C
Words + Image features	IG(1000) for trees	Logistic + Neural Netwo
Words + Image features	IG(1000) for trees	Logistic + Neural Netwo
Words + Image features + Image RGB	IG(1000); Face detection + PCA(100)	Logistic + Neural Network + Ensemble
Words + Image features + Image RGB	IG(1000); face-HOG	Logistic + Neural Network + Ensem
Words + Image features + Image RGB	IG(1000); face/eyes/nose-HOG	Logistic + Neural Network + Ensem

Table 2: Experimental results of ensemble classifiers