Weird News Classification & Ranking

IRE-Project Phase-1

Group: 13
Mentor: Vijayasaradhi Indurthi

Problem Statement

Figuring out the weirdness score and ranking weird or odd news stories.

What we promised

- Classification of the news i.e. whether it is weird or not.
- Comparison with existing Machine Learning techniques to provide a survey of the performance of various models for this task.
- Providing a novel Deep Learning architecture for this classification task.

Classification of News

Classifiers used:

- ➔ Naive bayes
- ➔ support vectors machines
- ➔ Random forest classifiers
- ➔ Gradient boosting classifiers
- ➔ Ada boosting classifiers
- ➔ Convolutional neural networks
- ➔ Decision tree classifier
- ➔ 3-layered-Perceptron
- ➔ LSTM
- ➔ Auto ML

Auto.sklearn

Auto-sklearn frees a machine learning user from algorithm selection and hyperparameter tuning. It leverages recent advances in Bayesian optimization, meta-learning and ensemble construction.

Gaussian Naive Bayes

Naive Bayes is a kind of classifier which uses the Bayes Theorem. It predicts membership probabilities for each class such as the probability that given record or data point belongs to a particular class. The class with the highest probability is considered as the most likely class.

Naive Bayes classifier assumes that all the features are unrelated to each other.

Support Vectors Machines

We plot each data item as a point in n-dimensional space (where n is number of features you have) with the value of each feature being the value of a particular coordinate. Then, we perform classification by finding the hyperplane that differentiate the two classes very well.

- **kernel:** radial bases function
- **C**: 1000 (moderate value to avoid overfitting)

Random Forest

It is a type of ensemble learning method, where a group of weak models combine to form a powerful model.

We grow multiple trees as opposed to a single tree in CART model. To classify a new object based on attributes, each tree gives a classification and we say the tree “votes” for that class. The forest chooses the classification having the most votes.

XGBoost

Works by weighting the observations, putting more weight on difficult to classify instances and less on those already handled well. New weak learners are added sequentially that focus their training on the more difficult patterns.Predictions are made by majority vote of the weak learners' predictions, weighted by their individual accuracy.

- **learning_rate**: 1.0 (learning rate shrinks the contribution of each tree by learning_rate)
- **n_estimators**:100 (the number of boosting stages to perform)
- **Max_depth**: nodes are expanded until all leaves are pure or until all leaves contain less than 2 samples (the maximum depth of the tree)

ADABoost

We assign equal weights to all the training examples and choose a base algorithm. At each step of iteration, we apply the base algorithm to the training set and increase the weights of the incorrectly classified examples. We iterate n times, each time applying base learner on the training set with updated weights. The final model is the weighted sum of the n learners.

- **n_estimators**: 100 (the maximum number of estimators at which boosting is terminated)

CNN

CNNs are biologically-inspired models.

CNNs have an associated terminology and a set of concepts that is unique to them, and that sets them apart from other types of neural network architectures.

- **solver**: ‘lbfgs’ is an optimizer in the family of quasi-Newton methods
- **alpha**: L2 penalty (regularization term) parameter (euclidean)
- **ngram_filters:** [2, 3, 4, 5] (for each filter we have 300 neurons)
- **hidden layer size**: (300, 300, 2) The ith element represents the number of neurons in the ith hidden layer.

Decision Tree

Each interior node corresponds to one of the input variables, there are edges to children for each of the possible values of that input variable. Each leaf represents a value of the target variable given the values of the input variables represented by the path from the root to the leaf.

- **min_samples_split**: 2 (the minimum number of samples required to split an internal node)

LSTM

It exploits the recurrent relation in the words present in the title to obtain a unified vector representation of the title to better distinguish wierd news titles from normal titles in the vector space.

- lstm_layers: (100, 100)
- Hidden_layers: (300, 2)

the ith value in the above notation denotes the number of neurons in the ith layer

Auto-ML

Auto-sklearn frees a machine learning user from algorithm selection and hyperparameter tuning. It leverages recent advances in Bayesian optimization, meta-learning and ensemble construction.

Experiment Results

Method	Precision	Recall
Naive Bayes	80	78
SVM		
Random Forest Classifier	77	64
Gradient Boosting Classifier	79	79
Ada Boosting Classifier	79	79
Decision Tree Classifier	78	79
3-layered-Perceptron	89	89
LSTM	91	91
CNN	90	90
auto-ML		

Plan for final Phase

The Ranking Task:

- Ranking of the weird news as per its weirdness score.
- Providing a way to rank a set of news based on their weirdness scores.
- Provide a metric that can be used to quantify the weirdness of a news.

Dataset for ranking evaluation:

- The current datasets do not provide a way to evaluate the ranking proposed by any algorithm. We plan to develop a good quality dataset that can be used as a gold standard for evaluation.

Thanks!

Pinkesh Badjatiya (201402002)
Anupam Pandey (20162118)
Nakul Vaidya (201501108)