Tejashree Khot and Sanjana Agarwal

ASSIGNMENT 4 - MACHINE LEARNING IN COMPUTATIONAL LINGUISTICS

We have used the majority class as the baseline.
Since some data points had more than one class, we used the last class of every data point.
The following tables depict the parameters used and the accuracy associated with it.

1) ARM.N

Baseline: 0.819

PARAMETERS	ACCURACY
scikit-learn LinearSVC penalty=**'l2'**, loss=**'hinge'**, dual=True, tol=0.0001, C=0.10, multi_class=**'ovr'**, fit_intercept=True, intercept_scaling=100, class_weight=**'balanced'**, verbose=0, random_state=None, max_iter=1740	0.842105263158
scikit-learn K Neighbors Classifier n_neighbors=5, weights=**'distance'**, algorithm=**'brute'**, leaf_size=20, metric=**'manhattan'**, p=3, metric_params=None, n_jobs=1	0. 81954887218
scikit-learn Multinomial Naive Bayes alpha=0.35, fit_prior=True, class_prior=None	0.887218045113
timbl -dIL -mN -k5 -N5557 -f ~/arm.n.train -t ~/arm.n.test -dIL: Linear Decay -mN: Numeric Distance -k5: '5' nearest neighbor - N5557: Feature set size is 5557	0.819549

2) DIFFICULTY.N
Baseline: 0.35

PARAMETERS	ACCURACY
scikit-learn LinearSVC penalty='**l2**', loss='**hinge**', dual=True, tol=0.0001, C=1.0, multi_class='**ovr**', fit_intercept=True, intercept_scaling=250, class_weight='**balanced**', verbose=0, random_state=None, max_iter=1500	0.545454545455
scikit-learn K Neighbors Classifier n_neighbors=3, weights='**distance**', algorithm='**auto**', leaf_size=30, metric='**manhattan**', metric_params=None, n_jobs=1	0.409090909091
scikit-learn Multinomial Naive Bayes alpha=1.0, fit_prior=True, class_prior=None	0.227272727273
timbl -dED -mN -k3 -f ~/difficulty.n.train -t ~/difficulty.n.test -dED: Exponential Decay -mN: Numeric Distance -k3: ‘3’ nearest neighbor	0.409091

3) INTEREST.N
Baseline: 0.42

PARAMETERS	ACCURACY
scikit-learn Linear SVC penalty=**'l2'**, loss=**'hinge'**, dual=True, tol=0.0001, C=1.0, multi_class=**'ovr'**, fit_intercept=True, intercept_scaling=250, class_weight=**'balanced'**, verbose=0, random_state=None, max_iter=2000	0.623655913978
scikit-learn K Neighbors Classifier n_neighbors=1, weights=**'uniform'**, algorithm=**'kd_tree'**, leaf_size=30, metric=**'minkowski'**, p=2, metric_params=None, n_jobs=1	0.505376344086
scikit-learn Multinomial Naive Bayes alpha=0.65, fit_prior=False, class_prior=None	0.688172043011
timbl -dED -mN -k5 -N5281 -f ~/interest.n.train -t ~/interest.n.test -dED: Exponential Decay -mN: Numeric Distance -k5: '5' nearest neighbor -N5281: Feature set size is 5281	0.473118

Feature extraction description:

1) We used 'Beautifulsoup' library as our HTML parser to extract information from the English.ls files.
2) Then, we created a dictionary of the context words from the training set and assigned each word a unique number.
3) This unique number represents the index of the word in the input vector.
4) Then we read the context for each instance and convert it to input vectors using the dictionary.

We couldn't run timbl for interest.n and arm.n because Number of Features exceeded the maximum number(2500).

In case of arm.n and difficulty.n, the best result for timbl is same as best result of scikit-learn's knn algorithm and comparable results for interest.n by both. Also, other (scikit-learn) algorithms are out performing knn(except in one case).

Among the scikit-learn algorithms, Linear SVC performed consistently well for all 3 words compared to K nearest neighbours. Naïve Bayes gave best results for interest.n and arm.n words but below baseline results for difficulty.n.