# BrainEE at SemEval-2019 Task 3: Ensembling Linear Classifiers for Emotion Prediction

**Vachagan Gratian**
Universität Stuttgart
vgratian@utopianlab.am

## Abstract

We present a homogeneous ensemble of linear perceptrons trained for emotion classification as part of the SemEval-2019 shared-task 3. The model uses a matrix of probabilities to weight the activations of the base-classifiers and makes a final prediction using the *sum rule*. The base-classifiers are multi-class perceptrons utilizing character and word n-grams, part-of-speech tags and sentiment polarity scores. The results of our experiments indicate that the ensemble outperforms the base-classifiers, but only marginally. In the best scenario our model attains an F-Micro score[1] of 0.672, whereas the base-classifiers attained scores ranging from 0.636 to 0.666.

## 1 Introduction

Our task is to detect emotions in multi-turn chat messages (see examples in table 1). The four emotion categories the model has choose from are `happy, sad, angry` and `others`. A major caveat of the task is the imbalance of class distribution in the dataset, as described in 4.1. The dataset, as well as the task itself are described in detail in (Chatterjee et al., 2019).

We choose to deploy ensemble of linear classifiers for this task, rather than a single model for a number of reasons. Firstly, given the inherent ambiguity of emotions (Brainerd, 2018) we expect that ensembles are better suited for any emotion prediction task. Secondly, it has been shown that ensembles are more immune to overfitting in similar tasks (Dong and Han, 2004). And finally, a single model trained on a large number of feature sets, tend to perform significantly worse than an ensemble where each model is trained on a different subset (or combinations) of feature types.

| Conversation | Emotion |
|---|---|
| A: Yes<br>B: How so?<br>A: Don't message me ever | angry |
| A: I am fine<br>B: I am good how is ur week<br>A: I am single | others |

Table 1: Two samples from the dataset with *angry* and *others* respectively as gold labels.

For this purpose, we deploy *BrainT*, a multi-class perceptron model utilizing word n-grams and POS-tags, built and trained for implicit emotion detection in Tweets (Gratian and Haid, 2018). In the current scenario, we extend the feature sets of *BrainT* with character n-grams and Sentiment polarity scores. We combine $n = 11$ and $n = 5$ classifiers into an ensemble model where a final prediction is made based on the activations. Our model also calculates a matrix of probabilities used to weigh the input activations. Each element in the matrix is the probability of a given node making correct prediction for a given emotion class. In the initial experiments the nodes are trained on the full train data. In the second group of experiments, nodes are assigned a random subsets of the train data separately. We hope that this will promote diversity in the base-classifiers and boost the performance of the ensemble.

The results of our experiments indicate that in both cases the ensemble outperforms the base-classifiers, however only slightly. In the following sections we describe the architecture of the model, the actual results on the SemEval shared-task. Finally we suggest ways to maximize the effectiveness of ensemble models as ideas for future work.

[1] This is a custom F-Micro score. See more details under 4.3 Evaluation

*Proceedings of the 13th International Workshop on Semantic Evaluation (SemEval-2019)*, pages 137–141
Minneapolis, Minnesota, USA, June 6–7, 2019. ©2019 Association for Computational Linguistics

## 2 Related Work

Ensemble learning aims at exploiting the ”shared knowledge” of multiple classifiers based on Statistical Learning theory. A theoretical analysis of ensemble learning using linear perceptrons, can be found in (Hara and Okada, 2005) and (Miyoshi et al., 2005). The authors demonstrate that the generalization error of ensemble learning depends on (hence, can be calculated from) two cosine measures: the similarity between the base-classifiers and the training data and the mutual similarity of the base-classifiers. In plain English, to maximize the performance of the ensemble model, we want to increase the accuracy of the base-classifiers but in such a way that we promote diversity in the base-classifiers.

A simple way of combining the base-classifiers is to take the average of their weights after training. A more common approach is to exploit the output activations using different techniques. In (Xia et al., 2011a) three such techniques are analyzed for sentiment classification: fixed combination, weighted combination and meta-classifier combination. The authors found fixed combination to be the weakest of all three, while weighted combination and meta-classifier added on average 3-4% improvement over the performance of the best base-classifier.

In our work, we deploy the weighted combination technique with the addition of a learned probability matrix, as described below.

## 3 The Model

### 3.1 Base-Classifiers

We use as base-classifier the linear perceptron model described in (Gratian and Haid, 2018) which reduces the task of multi-class prediction into $|Y|$ binary classification problems (where $|Y|$ is the number of emotion classes) following the ”one-against-all” approach described by Xia et al. (2011b).

The output of each base-classifier is a vector $\alpha$ of size $|Y|$ corresponding to the number of emotions. Before passing this vector to the ensemble model, each activation $\alpha_y$ is calibrated as follows:

$$\hat{\alpha}_y = \frac{\alpha_y}{\sum\limits_{\hat{y} \in Y} \alpha_{\hat{y}}}$$

By doing so, each $\hat{\alpha}_y$ can be treated as a confidence level of the node that the instance $x_i$ expresses the emotion class $y$. The advantage of this approach is that even when the true emotion class is not the one predicted by the node (i.e., it is not the highest activation), it can still contribute to the true class being predicted by the ensemble if it has a positive value.

### 3.2 Ensemble

The ensemble model $M$ represents a matrix of probabilities of size $n \times m$:

$$M = \begin{pmatrix} \varphi_{1,1} & \varphi_{1,2} & \cdots & \varphi_{1,m} \\ \varphi_{2,1} & \varphi_{2,2} & \cdots & \varphi_{2,m} \\ \vdots & \vdots & \ddots & \vdots \\ \varphi_{n,1} & \varphi_{n,2} & \cdots & \varphi_{n,m} \end{pmatrix}$$

The $ij$-th element of this matrix is the probability that $i$-th node’s prediction for $j$-th class is correct. This parameter is initialized to $1$ and is learned during training as:

$$\varphi_{i,j} = \frac{|R_{ij}|}{|R_{ij}| + |R'_{ij}|}$$

where $|R_{ij}|$ and $|R'_{ij}|$ are respectively the correct and incorrect predictions of node $i$ for class $j$ during training. This probability value, thus, corresponds to the Precision metric.

The input of the model is a matrix of equal size: those are the activations of $n$ nodes, each a vector of size $m$ (or $|Y|$). We weigh these activations by the probabilities learned by the model by taking the Hadamard product of the two matrices. The final prediction is made by the ensemble following the *sum rule* as described by (Xia et al., 2011b). The predicted class $\hat{y}$ is the one that has the highest sum of weighted activations:

$$\hat{y} = \text{argmax}_{i=1}^{m} \sum_{j=1}^{n} \varphi_{i,j}\, \hat{a}_{i,j}$$

## 4 Experiments

### 4.1 Dataset

The dataset we use is provided by the SemEval2019 shared-task 3 and is described in detail by (Chatterjee et al., 2019). It contains a train set with 30,160 and a test set with 5,509 conversations. In both sets the emotion class *others* is disproportionately overrepresented. Moreover, there

| Angry | Sad | Happy | Others | Total |
|---|---|---|---|---|
| 5,506 | 5,463 | 4,243 | 14,948 | 30,160 |
| 18.3% | 18.1% | 14.1% | 49.6% | 100% |

Table 2: Class distribution in the train dataset.

| Angry | Sad | Happy | Others | Total |
|---|---|---|---|---|
| 298 | 250 | 284 | 4,677 | 5,509 |
| 5.4% | 4.5% | 5.2% | 84.9% | 100% |

Table 3: Class distribution in the test dataset.

is a significant difference between the class distributions in the train and test datasets as can be seen in tables 2 and 3 imposing an additional challenge for the classification task.

The evaluation metric of the shared task is a custom F-micro measure which takes only into account the three emotion classes (`happy`, `angry`, `sad`) and disregards the overrepresented class `others`.

### 4.2 Preprocessing

As in the previous experimental setup, we apply minimal preprocessing. We don't normalize tokens and don't filter stopwords as this proved to decrease system performance in our previous experiments. We treat the 3 turns in each conversation as one stream of tokens by concatenating them using the special token $\langle STOP\rangle$.

### 4.3 Features

Our feature types are word and character n-grams, as well as POS tags extracted with the NLTK part-of-speech tagger and polarity scores from the `Sentiment Classification using WSD` library [2].

The word n-grams include unigrams, bigrams, trigrams and tetragrams where one token is replaced with the placeholder $\langle SKIP\rangle$ tag as this feature type proved to be highly efficient in our previous experiments.

The list of feature types used in our experiments is in table 4.

### 4.4 Experimental Setup

We assign each node 2 to 4 feature types. In the preparatory stage of the experiments we train nodes with different combinations of the feature types and select the 11 highest ranking nodes. Table 6 lists these nodes.

| Feature Set | Description |
|---|---|
| 1GR<br>2GR<br>3GR<br>4GR-S1 | word n-grams |
| 1CH<br>2CH<br>3CH | character n-grams |
| POS<br>SENTA | Lexicon-based |

Table 4: The feature types utilized by the base-classifiers.

| | False Positives | False Negatives |
|---|---|---|
| Happy, Sad, Angry | 0.8 | 1.5 |
| Others | 0.1 | 0.5 |

Table 5: Learning rates

To overcome overrepresantation of the class `others` we apply a lower learning rate for this class. We furthermore apply a higher learning rate for false negatives than false positives, since in the preparatory experiments all nodes showed a significantly lower Recall than Precision. Table 5 lists those learning rates.

Finally, we test the ensemble model in two experimental setups: *uniform learning* and *distributed learning*. In the first scenario, the entire train data is used to train the 11 nodes. Either all 11 node activations are passed to the ensemble or only those of the 5 highest performing nodes. In the second scenario, each node is assigned and trained on a random 50% subset of the train data.

For all our experiments we choose the number of epochs to be 60.

## 5 Results

### 5.1 Uniform Learning

In all of our experiments the ensemble performs only slightly better than the best performing node(s). The results of the experiment with uniform training are in Table 6. We observe that reducing the number of nodes from 11 to 5, decreases Precision of the ensemble, but increases Recall, however in both cases the difference is insignificant.

[2] The library is free-software and is available online: `https://github.com/kevincobain2000/sentiment_classifier`

| Node | Precision | Recall | F-micro |
|---|---|---|---|
| 1GR_3GR_3CH_POS | 0.604 | 0.672 | 0.636 |
| 2GR_4GR-S1_3CH_POS | 0.627 | 0.675 | 0.65 |
| 2GR_4GR-S1_3CH | 0.622 | 0.69 | 0.654 |
| 1GR_2GR | 0.613 | 0.706 | 0.656 |
| 2GR_SENTA | 0.661 | 0.661 | 0.661 |
| 2GR_3CH_SENTA | 0.649 | 0.677 | 0.662 |
| 1GR_2GR_POS | 0.632 | 0.695 | 0.662 |
| 1GR_2GR_SENTA | 0.636 | 0.692 | 0.663 |
| 1GR_2GR_3CH_SENTA | 0.632 | 0.697 | 0.663 |
| 1GR_2GR_1CH | 0.638 | 0.696 | 0.666 |
| 1GR_2GR_3CH | 0.631 | 0.704 | 0.666 |
| **ENSEMBLE_N=5** | **0.640** | **0.700** | **0.672** |
| **ENSEMBLE_N=11** | **0.649** | **0.694** | **0.671** |

Table 6: Results for Exp 1 with 11 nodes and uniform training.

| Node | Precision | Recall | F-micro |
|---|---|---|---|
| 1GR_3GR_3CH_POS | 0.594 | 0.614 | 0.604 |
| 2GR_SENTA | 0.635 | 0.579 | 0.606 |
| 2GR_4GR-S1_3CH_POS | 0.6 | 0.629 | 0.614 |
| 2GR_4GR-S1_3CH | 0.589 | 0.649 | 0.617 |
| 2GR_3CH_SENTA | 0.639 | 0.626 | 0.633 |
| 1GR_2GR_3CH | 0.607 | 0.665 | 0.635 |
| 1GR_2GR_1CH | 0.615 | 0.669 | 0.641 |
| 1GR_2GR_SENTA | 0.622 | 0.666 | 0.643 |
| 1GR_2GR_3CH_SENTA | 0.623 | 0.663 | 0.643 |
| 1GR_2GR | 0.616 | 0.675 | 0.644 |
| 1GR_2GR_POS | 0.623 | 0.668 | 0.645 |
| **ENSEMBLE_N=11** | **0.648** | **0.666** | **0.657** |

Table 7: Results for Exp 2 with 11 nodes and distributed training.

We also observe that while the ensemble outperforms the nodes in the F-micro measure, it has a lower Precision and Recall than at least one of the nodes.

### 5.2 Distributed Learning

In the second experiment each node is trained on a 50% random subset of the train data. We observe a drop in the performance of both the ensemble and the nodes, although the ensemble outperforms the nodes with a slightly larger margin.

Compared to the results of uniform learning, we see that the ensemble has roughly the same Precision, but a lower Recall. However when we compare the performance of the ensemble with the base-classifiers, we see that the ensemble now has now a higher Precision score than any of the nodes. This indicates that the ensemble benefits more from the ”shared knowledge” of the base-classifiers.

## 6 Discussion

The goal of our experiments was to build an ensemble that makes better predictions than any of the base-classifiers individually. While the results of our experiments prove this to be a success, they also indicate that the ensemble exploits the strengths of the nodes weakly. For most of the emotion classes, the ensemble underperforms at least one of the nodes.

This disparity is especially vivid in the Recall measure. We presume that this due to the fact that the probabilities matrix learned by the model reflects only Precision, not Recall. As a future improvement to the model, we could adapt the probabilities to reflect Recall as well.

## 7 Future Work

In our future work we want to adopt a different approach to ensemble learning. Firstly, we think an important starting point should be a concrete estimation of the ensemble's upper bound performance given $n$ base-classifiers. This can then serve as banchmark to evaluate the actual performance of an ensemble model. In most, if not all, real-world situations, the probability that a node $n_i$ makes a correct prediction for an instance $x_j$ will always be conditional to the probability of another node $n_j$. This means that the upper boundary of the ensemble model depends on the conditional probabilities of its nodes.

This implies that in our future work we will describe ensemble learning as the task to minimize joint entropy of the base-classifiers in addition to maximizing accuracy.

## 8 Conclusion

In this paper we describe an ensemble model trained for emotion classification. We evaluate our model on uniform and distributed learning of the train data. The results of the experiments indicate that while the model outperforms the strongest model, it benefits weakly from the strengths and variance of the base-classifiers.

## References

C. J. Brainerd. 2018. The emotional-ambiguity hypothesis: A large-scale test. *Psychological Science*, 29(10):1706–1715. PMID: 30130163.

Ankush Chatterjee, Kedhar Nath Narahari, Meghana Joshi, and Puneet Agrawal. 2019. Semeval-2019 task 3: Emocontext: Contextual emotion detection in text. In *Proceedings of The 13th International Workshop on Semantic Evaluation (SemEval-2019)*, Minneapolis, Minnesota.

Yan-Shi Dong and Ke-Song Han. 2004. A comparison of several ensemble methods for text categorization. pages 419– 422.

Vachagan Gratian and Marina Haid. 2018. Braint at iest 2018: Fine-tuning multiclass perceptron for implicit emotion classification. In *Proceedings of the 9th Workshop on Computational Approaches to Subjectivity, Sentiment and Social Media Analysis*, pages 243–247. Association for Computational Linguistics.

Kazuyuki Hara and Masato Okada. 2005. Ensemble learning of linear perceptrons: On-line learning theory. *Journal of The Physical Society of Japan - J PHYS SOC JPN*, 74:2966–2972.

Seiji Miyoshi, Kazuyuki Hara, and Masato Okada. 2005. Analysis of ensemble learning using simple perceptrons based on online learning theory. *Physical review. E, Statistical, nonlinear, and soft matter physics*, 71:036116.

Rui Xia, Chengqing Zong, and Shoushan Li. 2011a. Ensemble of feature sets and classification algorithms for sentiment classification. *Information Sciences*, 181(6):1138 – 1152.

Rui Xia, Chengqing Zong, and Shoushan Li. 2011b. Ensemble of feature sets and classification algorithms for sentiment classification. *Inf. Sci.*, 181:1138–1152.