COMP90049 Project 2 Report: Which emoji is missing?

1 Introduction

The project is to analyse the effectiveness of supervised Machine Learning methods on the problem that predicting which emoji should be used in a tweet. In Machine Learning perspective, to determine the emoji used in a text is a classification problem.

The Machine Learning methods that we are going to analyse are Naive Bayes Classifier, Decision Stump (one-level Decision Tree) Classifier, and TensorFlow Deep Neural Network (DNN) Classifier. This report is mainly focused on DNN Classifier, whilst other two methods are only used for comparison and critical analysis.

In general, we are going to use a training dataset to train the DNN model with a text embedding module (will explain later), and using the trained model to predict results on a test dataset. We will be using TensorFlow and some Machine Learning packages in Python. The evaluation of DNN Classifier will be done accordingly within a package. Other two methods will be applying in a similar process but using Weka.

2 Problem

Which emoji is missing? Nowadays, on the Internet, most people are texting and tweeting with emojis. It gives a plain text an emotional feature. We are going to predict an emoji from a given plain text in a tweet. This emoji problem is similar to sentiment analysis, but in a discrete way; instead of a continues value represents how positive of a text, we categories a text to a specific emoji class e.g. Happy, Sad etc.

In such text classification problem, text is the only dependency of that problem. In order to train our Machine Learning model effectively, we should carefully define features and do a lot of works on data preprocessing to obtain as much as information we can from a text.

3 Dataset

The dataset contains three set of data: training, development and test sets, as well as some given feature representations such as top10 and most100.

4 Methods

.....

5 Evaluation

.....

6 Analysis

.....

7 Conclusion

.....

References

[1] Saphra, Naomi; Lopez, Adam *Evaluating Informal-Domain Word Representations With UrbanDictionary* 2016: Proceedings of the 1st Workshop on Evaluating Vector Space Representations for NLP.

[2] Levenshtein, Vladimir I. *Binary codes capable of correcting deletions, insertions, and reversals* 1966: Soviet Physics Doklady.

[3] Navarro, Gonzalo *A guided tour to approximate string matching* 2001: ACM Computing Surveys.

[4] Broder, Andrei Z.; Glassman, Steven C.; Manasse, Mark S.; Zweig, Geoffrey *Syntactic clustering of the web* 1997: Computer Networks and ISDN Systems.

[5] Kondrak, Grzegorz *N-gram similarity and distance* 2005: International Symposium on String Processing and Information Retrieval.