SSN_NLP at SemEval-2020 Task 7: Detecting funniness level using traditional learning with sentence embeddings

Kayalvizhi S
SSN College of Engineering
kayalvizhis@ssn.edu.in

Thenmozhi D
SSN College of Engineering
theni_d@ssn.edu.in

Aravindan Chandrabose
SSN College of Engineering
aravindanc@ssn.edu.in

Abstract

Assessing the funniness of edited news headlines task deals with estimating the humorness in the headlines edited with micro-edits. This task has two sub-tasks in which one has to calculate the mean predicted score of humor level and other deals with predicting the best funnier sentence among given two sentences. We have calculated the humorness level using microtc and predicted the funnier sentence using microtc, universal sentence encoder classifier, many other traditional classifiers that use the vectors formed with universal sentence encoder embeddings, sentence embeddings and majority algorithm within these approaches. Among these approaches, microtc with 6 folds, 24 processes and 3 folds, 36 processes achieve the least Root Mean Square Error for development and test set respectively for subtask 1. For subtask 2, Universal sentence encoder classifier achieves the highest accuracy for development set and Multi-Layer Perceptron applied on vectors vectorized using universal sentence encoder embeddings for the test set.

1 Introduction

Assessing the Funniness of Edited News Headlines (Hossain et al., 2020) task of Semeval-2020 deals with estimating the funniness of news headlines that have been modified by humans using a micro-edit to make them funny. The goal of the task is to make the machine predict whether the short edits on the given text makes them funny or not. Thus, the system has to assess the intensity of humorness in the edited sentence. This system has various applications including humor generation where such a system can be used in a generate-and-test scheme to generate many potentially humorous texts and rank them in terms of funniness. This task has two sub tasks that include a regression task, in which the mean funniness of the edited headline has to be predicted and a classification task, in which the funnier of the two edited headlines has to be predicted.

2 Related Work

Similar task called Humor Analysis based on Human Annotation (HAHA) has been part of IberLEF 2018 (Castro et al., 2018) and IberLEF 2019 (Chiruzzo et al., 2019) which deals with automatic detection and automatic rating of humor in Spanish tweets. Different methodologies have been reported in these tasks. Convolutional RNN (Giudice, 2019) have been used, in which input tweet is represented as a list with fixed length of sparse vectors and each vector of the list represents an individual character of the tweet and contains flags whose values are either 0 or 1 is fed to the neural network whose output was rounded to 0 or 1 to get a binary value and for the funniness score prediction subtask it was multiplied by five to get a value between 0 and 5. In a BERT based approach (Mao and Liu, 2019; Ismailov, 2019) a mutli-lingual model with modified hidden layers and output layers have been used. In Bi-directional LSTM approach (Garain, 2019), the manually-extracted features are passed through a neural network with sigmoid activation function. Gaussian process (Miller et al., 2019) have also been done in which vectors are obtained using pretrained Spanish Twitter embeddings and then trained and tested in the Gaussian process regressor model. Ingeotec (Ortiz-Bejar et al., 2018; Ortiz-Bejar et al., 2019) have made use of

Proceedings of the 14th International Workshop on Semantic Evaluation, pages 865–870
Barcelona, Spain (Online), December 12, 2020.

EvoMSA, a multilingual sentiment classifier and MicroTc, a minimistic classifier tool that uses Support Vector Machine.

3 Dataset Description

Each subtask has a dataset with the instances described in Table 1. The dataset (Hossain et al., 2019) also includes some additional train set of funniness instances.

Sub task 1 dataset :
id $< tab >$ original text $< tab >$ edit $< tab >$ grade $< tab >$ Meangrade
Sample instance :
76 $< tab >$ US imposes metal tariffs on key $< allies/ >$ $< tab >$ holes $< tab >$ 10000 $< tab >$ 0.2

Inference :
edited sentence : "US imposes metal tariffs on key holes"
score : 0.2

Sub task 2 dataset :
id $< tab >$ original sentence 1 $< sep >$ sentence 1 edit $< sep >$ sentence 1 grade $< tab >$ sentence 1 mean grade $< tab >$ original sentence 2 $< tab >$ sentence 2 edit $< tab >$ sentence 2 grade $< tab >$ sentence 2 mean grade $< tab >$ label
Sample instance :
10722-3702 $< tab >$ " I 'm done " : Fed up with $< California/ >$, some conservatives look to Texas $< tab >$ pancakes $< tab >$ 10110 $< tab >$ 0.6 $< tab >$ " I 'm done " : Fed up with $< California/ >$, some conservatives look to Texas $< tab >$ life $< tab >$ 2 $< tab >$ 0.4 $< tab >$ 1

Inference:
edited sentence 1 : " I 'm done : Fed up with pancakes , some conservatives look to Texas"
score of sentence 1 : 0.6
edited sentence 2 : " I 'm done : Fed up with life, some conservatives look to Texas"
score of sentence 2 : 0.4
label : 1

Set	**Task-1**	**Task-2**
Training set	9382	9653
Additional training set	8249	1959
Development set	2420	2356
Test set	2961	3025

Table 1: Data set

4 Methodology

4.1 Sub task-1 : Regression

The first subtask deals with predicting the humor level in the edited news headline. We have used MicroTc (Tellez et al., 2018) for predicting the funniness score with varying parameters such as number of folds and number of processes. MicroTC is a Machine Learning approach that classifies data based on SVM linear kernel. It is a combinatorial function of four steps namely pre-processing, tokenizing, vector representation and classification using SVM linear kernel. Pre-processing includes removal of tags, punctuations, urls and converting them to lower case. Tokenization is done by skip-gram and n-gram models and they are converted to vectors by TF-IDF, TF methods and concatenating the maximum and minimum filters and then classifying the data. The edited sentences are given to the model in json file format. The format is as follows:

Sample input in training json file:
{“text”: “US imposes metal tariffd on key holes ”, “decision_function”: 0.2, “klass”: “1”}
{“text”: “For The First Time In Years Shops Have More Guns Than bullets ”, “decision_function”: 0.0, “klass”: “0”}

Sample output:
{“text”: “For The First Time In Years Shops Have More Guns Than bullets ”, “decision_function”: -1.0462107867013373, “klass”: “0”, “predicted”: “0”}
{“text”: “The pursuit of happiness : The American cultural case for a universal basic remote ”, “decision_function”: 2.267572997601439, “klass”: “1”, “predicted”: ”1”}

In the input format, text represents the text, decision function represents the mean grade and klass represents the label (whether funny or not funny). In the output format, the decision function represents the mean grade which was considered as the funniness score (i.e.) output.

4.2 Sub task-2 : Predict the funnier

This subtask was considered as a classification task since this subtask is to predict the funnier among them. Thus, the unique edited headlines are extracted from the training set and are trained with their corresponding funniness score. The methodology used are explained below:

4.2.1 MicroTc

In this approach, sentences from the training set are trained with their corresponding score of the sentence as its class. For implementation, microtc is installed and the json file is given as input to the model. Initially, hparams file is created followed by the model for which the input file must be given which produces the output file. MicroTC is a Machine Learning approach that classifies data based on SVM linear kernel. It is a combinatorial function of four steps namely pre-processing, tokenizing, vector representation and classification using SVM linear kernel. Pre-processing includes removal of tags, punctuations, urls and converting them to lower case. Tokenization is done by skip-gram and n-gram models and they are converted to vectors by TF-IDF, TF methods and concatenating the maximum and minimum filters and then classifying the data. The data format is the same as that of sub-task 1 and then predicts the funnier by comparing the decision function obtained.

4.2.2 Traditional Classifiers

In this method, the sentences are vectorized initially and then classified using traditional classifiers such as Support Vector Machine (SVM), Multi-layer Perceptron (MLP), Decision Tree (DT), K-Nearest Neighbour Classifier (KNC), Random Forest (RF), AdaBoost Classifier(ABC) and Gaussian Naive Bayes Classifier (GNB). For vectorization, two embeddings namely universal sentence encoder embeddings and sentence embeddings have been used which are explained below.

Universal Sentence Encoder Embeddings

In this approach, the edited headlines are vectorized using universal sentence encoder embeddings. The vector size is ‘512’. The sentences are initially read from the text file which is split and fed as a list of sentences to the module of universal sentence encoder [1] using tensorflow hub. Thus, embeddings for each sentence are generated.

[1] https://tfhub.dev/google/universal-sentence-encoder/2

Sentence Embeddings
In this approach, the edited headlines are vectorized using Sentence embeddings of BERT (Reimers and Gurevych, 2019). The sentences are encoded to vectors using the "bert-base-nli-mean-tokens" model of sentence transformers whose vector size is '768'.

After vectorizing, the vectors are classified using different classifiers such as SVM, MLP, Decision Tree (DT), K-Nearest Neighbour Classifier (KNC), Random Forest (RF), AdaBoost Classifier(ABC) and Gaussian Naive Bayes Classifier (GNB).
Classification of the vectors gives us the scores of two sentences which are compared. Then a funnier sentence is labelled by comparing the score.

4.2.3 Majority voting algorithm

Majority voting algorithm is applied between the classifiers approach and a microtc approach. The funnier one is decided by the result of the majority voting algorithm.

4.2.4 Universal Sentence Encoder Classifier

In this approach, the edited headlines are classified using universal sentence encoder (Cer et al., 2018). The text file with id, score and text are given as input to the model.

Format:
id $< tab >$ score (mean grade) : text

Sample input:
10920 $< tab >$ 1.2:Gene Cernan Last Dancer on the Moon Dies at 82
9866 $< tab >$ 0.8:Gene Cernan Last Astronaut on the Moon impregnated at 82

The model comprises of an input layer, an embedding layer followed by two dense layers. The input text is given to the input layer which is vectorized by universal sentence encoder embedding in the embedding layer and then given to a dense layer with 'relu' as activation function and 256 nodes followed by a dense layer with 'softmax' activation layer. The model was trained for 150 epochs with '32' as its batch size to predict the score.

5 Results

All the results of the development set and test set obtained were submitted to both the practice session and evaluation phase in the same order as in the table but the last ones in the table took place in the leaderboard. The results of sub-task 1 and sub-task 2 have been tabulated in Table 2 and Table 3 respectively.
From the Table 2, it is clear that microtc with varying number of folds and processes have been submitted among which microtc with 3 folds and 36 processes performed better in evaluation phase with least RMSE of 0.8447 and microtc with 6 folds, 24 processes and 8 folds, 24 processes have same RMSE of 0.7664 which is least among other combinations of folds and processes.

Number of folds	Number of processes	Dev set RMSE	Test set RMSE
3	36	0.7672	**0.8447**
6	24	**0.7664**	0.8452
8	24	**0.7664**	0.8727
3	24	0.8447	1.3502

Table 2: Sub-task 1 Evaluation results

Embeddings	Classifier	Accuracy	
		Dev set	**Test set**
Universal Embeddings	SVM	0.5336	0.5312
	MLP	0.5345	**0.5620**
	DT	0.5335	0.5030
	KNN	0.5221	0.5277
	RF	0.5312	0.5289
	ABC	0.5202	0.5068
	GNB	0.5259	0.5345
Sentence Transformers	SVM	0.5388	0.5403
	MLP	0.5416	0.5095
	DT	0.5250	0.5117
	KNN	0.5026	0.5091
	RF	0.5202	0.4497
	ABC	0.5364	0.4996
	GNB	0.5283	0.5178
Majority Algorithm		0.5531	0.5338
MicroTc		0.5455	0.5566
Ensemble		0.5469	0.5387
USE		**0.5764**	0.5376

Table 3: Sub-task 2 Evaluation results

For the sub-task 2, the universal sentence encoder classifier acquired the highest accuracy of 0.5764 for the development set and universal sentence embeddings vector classified using Mulit-Layer Perceptron achieves the best accuracy of 0.5620 which is shown in Table 3. But, since the last submission is considered as the system's result, our system achieves 0.5367 as the accuracy which is the universal sentence encoder classifier's result.

6 Conclusion

Assessing the funniness of edited headlines have been approached using different traditional approaches that include microtc, various classifiers namely MLP, SVM, Decision Tree, KNN, Random Forest, Gaussian Naive Bayes and AdaBoost Classifier that classifies the sentences vectorized using universal sentence encoder embeddings and sentence transformers, an ensemble method method of BERT classifier and microtc, universal sentence encoder classifier and a majority algorithm among them. For subtask 1, microtc with 6 folds, 24 processes and microtc with 3 folds, 36 processes achieve the least Root Mean Square Error for development set and test set respectively. For subtask 2, Universal sentence encoder classifier achieves the highest accuracy for development set and Multi-Layer Perceptron applied on vectors vectorized using universal sentence encoder embeddings for the test set. The performance may be further improved by using other embeddings and by fine tuning the microtc with other parameters.

Acknowledgments

We would like to thank HPC laboratory and DST-SERB for providing the resources.

References

Santiago Castro, Luis Chiruzzo, and Aiala Rosá. 2018. Overview of the haha task: Humor analysis based on human annotation at ibereval 2018. In *IberEval@ SEPLN*, pages 187–194.

Daniel Cer, Yinfei Yang, Sheng yi Kong, Nan Hua, Nicole Limtiaco, Rhomni St. John, Noah Constant, Mario Guajardo-Cespedes, Steve Yuan, Chris Tar, Yun-Hsuan Sung, Brian Strope, and Ray Kurzweil. 2018. Universal sentence encoder.

Luis Chiruzzo, S Castro, Mathias Etcheverry, Diego Garat, Juan José Prada, and Aiala Rosá. 2019. Overview of haha at iberlef 2019: Humor analysis based on human annotation. In *Proceedings of the Iberian Languages Evaluation Forum (IberLEF 2019). CEUR Workshop Proceedings, CEUR-WS, Bilbao, Spain (9 2019)*.

Avishek Garain. 2019. Humor analysis based on human annotation (haha)-2019: Humor analysis at tweet level using deep learning. In *Proceedings of the Iberian Languages Evaluation Forum (IberLEF 2019). CEUR Workshop Proceedings, CEURWS, Bilbao, Spain (9 2019)*.

Valentino Giudice. 2019. Aspie96 at haha (iberlef 2019): Humor detection in spanish tweets with character-level convolutional rnn. In *Proceedings of the Iberian Languages Evaluation Forum (IberLEF 2019). CEUR Workshop Proceedings, CEUR-WS, Bilbao, Spain*.

Nabil Hossain, John Krumm, and Michael Gamon. 2019. "president vows to cut <taxes> hair": Dataset and analysis of creative text editing for humorous headlines. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 133–142, Minneapolis, Minnesota, June. Association for Computational Linguistics.

Nabil Hossain, John Krumm, Michael Gamon, and Henry Kautz. 2020. Semeval-2020 Task 7: Assessing humor in edited news headlines. In *Proceedings of International Workshop on Semantic Evaluation (SemEval-2020)*, Barcelona, Spain.

Adilzhan Ismailov. 2019. Humor analysis based on human annotation challenge at iberlef 2019: First-place solution. In *Proceedings of the Iberian Languages Evaluation Forum (IberLEF 2019). CEUR Workshop Proceedings, CEUR-WS, Bilbao, Spain (9 2019)*.

Jihang Mao and Wanli Liu. 2019. A bert-based approach for automatic humor detection and scoring. In *Proceedings of the Iberian Languages Evaluation Forum (IberLEF 2019). CEUR Workshop Proceedings, CEUR-WS, Bilbao, Spain (9 2019)*.

Tristan Miller, Erik-Lân Do Dinh, Edwin Simpson, and Iryna Gurevych. 2019. Ofai–ukp at haha@ iberlef2019: Predicting the humorousness of tweets using gaussian process preference learning. In *Proceedings of the Iberian Languages Evaluation Forum (IberLEF 2019). CEUR Workshop Proceedings, CEUR-WS, Bilbao, Spain (9 2019)*.

José Ortiz-Bejar, Vladimir Salgado, Mario Graff, Daniela Moctezuma, Sabino Miranda-Jiménez, and Eric S Tellez. 2018. Ingeotec at ibereval 2018 task haha: μtc and evomsa to detect and score humor in texts. In *Proceedings of the Third Workshop on Evaluation of Human Language Technologies for Iberian Languages (IberEval 2018) co-located with 34th Conference of the Spanish Society for Natural Language Processing (SEPLN 2018)*.

José Ortiz-Bejar, Eric Tellez, Mario Graff, Daniela Moctezuma, and Sabino Miranda'Jiménez. 2019. Ingeotec at iberlef 2019 task haha. In *Proceedings of the Iberian Languages Evaluation Forum (IberLEF 2019). CEUR Workshop Proceedings, CEUR-WS, Bilbao, Spain (9 2019)*.

Nils Reimers and Iryna Gurevych. 2019. Sentence-bert: Sentence embeddings using siamese bert-networks. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing*. Association for Computational Linguistics, 11.

Eric S. Tellez, Daniela Moctezuma, Sabino Miranda-Jiménez, and Mario Graff. 2018. An automated text categorization framework based on hyperparameter optimization. *Knowledge-Based Systems*, 149:110–123.