Noname manuscript No.
(will be inserted by the editor)

Insert your title here

Do you have a subtitle? If so, write it here

Bobak Farzin[1], Nombre Apellidos2[2]
[1]**Universidad o lugar de trabajo**
[2]**Universidad o lugar de trabajo**
bfarzin@gmail.com

Received: 15 June 2019 / Accepted: date

Abstract Insert your abstract here. Include keywords, PACS and mathematical subject classification numbers as needed.

Keywords First keyword · Second keyword · More

1 Introduction

Included citations as necessary: [Jr(2006)]

Contribution Our contribution with this work is..

2 Task and Dataset Description

The *Humor Analysis based on Humor Annotation (HAHA)* competition asked for analysis of two tasks in the Spanish language based on a coprpus of publicly collected data [Castro et al.(2018)Castro, Chiruzzo, Rosá, Garat, and Moncecchi]:

- **Task1: Humor Detection**:Detemine if a tweet is humorous. System ranking is based on F1 score which will balance precision and accuracy.
- **Task2: Funniness Score**:If humorous, what is the average humor rating of the tweet. System ranking is based on RMSE.

HAHA dataset includes labeled data for 24,000 tweets and a test set of 6,000 tweets (80%/20% train/test split.) Each record includes the raw tweet text (including accents and emoticons) and a True/False labels as well as a "Funninness Score" that is the average of the 1 to 5 start votes cast. Examples and data can be found on the CodaLab competition webpage[1].

Address(es) of author(s) should be given

[1] http://competitions.codalab.org/competitions/22194/

3 System Description

We generally follow the method of ULMFiT [Howard and Ruder(2018)]

1. Train a language model (LM) on a large corpus of data
2. Fine-tune the LM based on the target task language data
3. Replace the final layer of the LM with a softmax or linear output layer and then fine-tune on the particular task at hand (classification or regression)

Below we will give more detail on each step and the parameters used to generate our system.

3.1 Data, Cleanup & Tokenization

3.2 Additional Data

For our initial training, we collected 475,143 tweets in the Spanish language using tweepy [**?**]. The vocabulary and frequency of terms, punctuation and vocabulary can be quite different from the standard Wikipedia corpus that is used to train.

We combined the labeled and un-labeled text data so that we have the largest corpus of language for our fine-tuning step.

3.3 Cleaning

We applied a list of default cleanup functions included in Fastai and added an additional one for this Twitter dataset.

- Add spaces between special chars (ie. `!!!` to `! ! !`)
- Remove useless spaces (remove more than 2 spaces in sequence)
- Replace repetition at the character level (ie. `grrrreat` becomes `g xxrep r 3 eat`)
- Replace repition at the word level (similar to above)
- Deal with ALL CAPS words replacing with a token and converting to lower case.
- **NEW:** Move all text onto a single line by replacing new-lines inside a tweet with a reserved word (ie. `\n` to `xxnl`)

Table 1 Please write your table caption here

first	second	third	fourth
number	number	number	
number	number	number	

Here is an example of replacing the original tweet with our parsed version.
Original:

```
Saber, entender y estar convencides que la frase \
#LaESILaDefendemosEntreTodes es nuestra linea es nuestro eje.\
#AlertaESI!!!!
Vamos por mas!!! e invitamos a todas aquellas personas que quieran \
se parte.
```

Cleaned up:

```
xxbos saber , entender y estar convencides que la frase \
# laesiladefendemosentretodes es nuestra linea es nuestro eje.\
xxnl  # alertaesi xxrep 4 ! xxnl vamos por mas ! ! ! e invitamos a \
todas aquellas personas que quieran se parte.
```

3.4 Tokenization

We used sentencepiece [Kudo and Richardson(2018)] to parse into sub-word units and reduce the possible out-of-vocabulary (OOV) terms in the data set. We selected a vocab size of 30,000 sub-word units and got 99.95% character coverage including emojis.

4 Training and Results

LM: 10% validation set for training

oversample minority class to balance for better training using SMOTE [Chawla et al.(2002)Chawla, Bowyer, Hall, and Kegelmeyer]

4.1 Random Seed as a Hyperparamter

5 Conclusion

Acknowlegements

References

[Castro et al.(2018)Castro, Chiruzzo, Rosá, Garat, and Moncecchi] Castro S, Chiruzzo L, Rosá A, Garat D, Moncecchi G (2018) A crowd-annotated spanish corpus for humor analysis. In: Proceedings of the Sixth International Workshop on Natural Language Processing for Social Media, pp 7–11

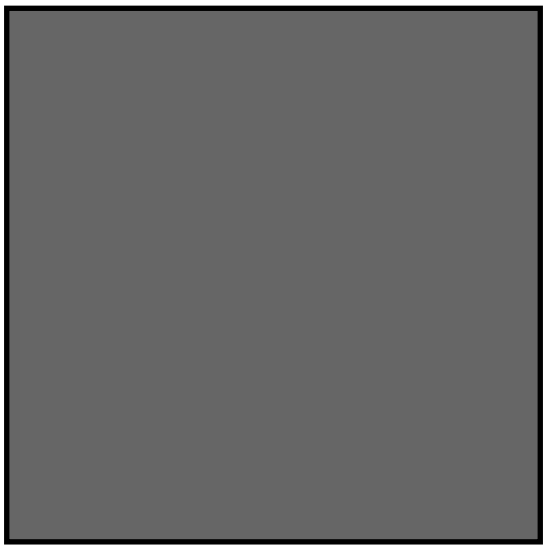

Fig. 1 Please write your figure caption here

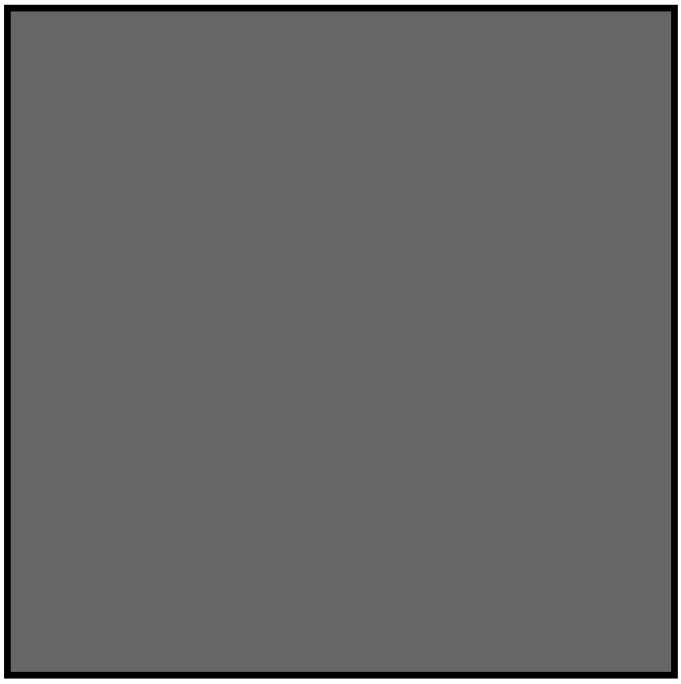

Fig. 2 Please write your figure caption here

[Chawla et al.(2002)Chawla, Bowyer, Hall, and Kegelmeyer] Chawla NV, Bowyer KW, Hall LO, Kegelmeyer WP (2002) Smote: Synthetic minority over-sampling technique. J Artif Int Res 16(1):321–357, URL http://dl.acm.org/citation.cfm?id=1622407.1622416

[Howard and Ruder(2018)] Howard J, Ruder S (2018) Fine-tuned language models for text classification. CoRR abs/1801.06146, URL http://arxiv.org/abs/1801.06146, 1801.06146

[Jr(2006)] Jr N (2006) My article

[Kudo and Richardson(2018)] Kudo T, Richardson J (2018) Sentencepiece: A simple and language independent subword tokenizer and detokenizer for neural text processing. CoRR abs/1808.06226, URL http://arxiv.org/abs/1808.06226, 1808.06226