Deep Modeling of Latent Representations for Twitter Profiles on Hate Speech Spreaders Identification Task.

Notebook for PAN at CLEF 2021

Roberto Labadie Tamayo[1], Daniel Castro Castro[1], and Reynier Ortega-Bueno[2]

[1] Universidad de Oriente, Cuba
roberto.labadie@estudiantes.uo.edu.cu, danielcc@uo.edu.cu
[2] PRHLT Research Center, Universitat Politècnica de València, Valencia Spain
rortega@prhlt.upv.es

Abstract Briefly describe the main ideas of your approach.

1 Paper Outline

The general architecture of our systems is modular, but in this work, over the individual tweets representation by means of a sentence encoder, we pay special attention to the way that profiles are modeled.

After the Introduction and the related works, we describe briefly the sentence encoder architecture as a transformer non-multilingual BERT pre-trained models, here, we must describe the intermediate task employed as well as the external data from Hateval 2018 to finetune this model. We also tested changing the ULM-Fit strategy for the use of adapters, whose performance is analyzed in the experiments and results section.

In Table 1 are shown the performance in the test set of different classifier models whose functionality lies on specific ways of modeling the authors feed as a deep representation of the textual information contained in their posts.

model	EN	ES	Avg
FCNN	0.81	0.73	0.77
Impostor Method (cosine)	**0.82**	**0.74**	**0.78**
Impostor (Deep-MetricL)	0.59	0.49	0.54
Graph Modeling (SGCN)	0.51	0.51	0.51
Sequential Modeling (Att-LSTM)	0.79	0.74	0.765

Table 1. Test-Set Evaluation trough Profile Modeling strategies

The first model in Table 1 (*FCNN*) is based in the idea of assuming each tweet within a profile as a token of a sequence, hence the whole profile is considered as a

fix-length sequence of 200 tokens, Over this sequence, we employ a Multimodal Gated Unit for defining which of this tokens provide more feasible evidence related to the hateful fashion in the author speech.

The second method (*Impostor Model - cosine*) simply combines the profile tweets by means of linear operations as mean or sum. This compressed representation is handled with a Machine Learning Method known as the Impostors Method, which gave to us our best results in English as well as in the Spanish language in competition. *(Here I propose to test this simple compressed representation with a shallow dense neural net, for comparing it fairly with respect to other representations)*.

The third method (*Impostor Model - DeepMetric L*) also receive the profile as a sequence, where employing an attention head each token is rated and combined, for feeding in a Siamese model whose output is used by the impostor method as a metric to compare two profiles for. Here the Metric learning is also an intermediate task.

Fourth Method models the whole profile in a graph structure, where each node represents a tweet and they are connected with each other since the data annotation does not provide individual information related to the tweets in particular. In the classification process each node shares and receive information from its neighbors, combining all this information by means of mean pooling and feeding a dense neural net.

Last but not least the (*Att-LSTM*) model receives again the profile as a sequence, this model is employed to compare the results with respect to the architecture used in PAN 2020 profiling task whose sentence encoder was based on Convolution and Recurrent Models.

All these models involve in a general way some difficulties or advantages for the classification process, I hypothesize for example, that in the Graph representation the way that connections are made, makes that in the message passing from one neighbor to another some important features vanish, especially if the profile belongs to a hate speech spreader but the hateful messages are just a few in their feed, or they contain some kind of soft-hate representation for saying it in some way (i.e the representation of hateful message is close to a neutral message representation).

Also, we think that the amount of data in terms of profiles, difficult this kind of deep learning method. Even when these models avoid the use of considerable hyper-parameters and tuning-time in the systems development from the ML method.

We Introduce the experiments comparing the performance of the 3 variants of language models finetuning (ULM-Fit, Adapters, Standard Fine tunning). We analyze the performance especially of the best two models, FCNN, Impostor Method. Here for the first case, we may determine which messages are receiving more importance the MGU feature combiner, as well as in the Impostor Method describe the use of specific, prototypes over just selecting random profiles, belonging to the hateful and not hateful class. Also,

we can compare the FCNN w.r.t the LSTM-Att model employed in PAN 2020 for this profiling task.

This Outline is just for you to know how big is this snowball :) and the directions we are taking, and for giving some recommendations in case that not all models be necessarily described. BTW, mire el titulo y me lo critica si algo.

2 Introduction

3 Related Works

4 Model Description

4.1 Profile Modeling

4.2 Impostor Method

5 Experiments and Results

6 Conclusion and Future Works