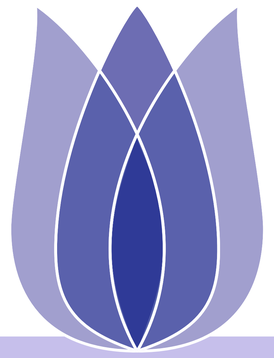
```
/burl@stx null def /BU.S  /burl@stx null def  def /BU.SS
currentpoint /burl@lly exch def /burl@llx exch def burl@stx
null ne burl@endx burl@llx ne BU.FL BU.S if if burl@stx
null eq  burl@llx dup /burl@stx exch def /burl@endx exch
def burl@lly dup /burl@boty exch def /burl@topy exch def
if burl@lly burl@boty gt /burl@boty burl@lly def if  def /BU.SE
currentpoint  /burl@ury exch def dup  /burl@urx exch def
/burl@endx exch def burl@ury burl@topy lt /burl@topy burl@ury
def if  def /BU.E BU.FL def /BU.FL burl@stx null ne BU.DF
if  def /BU.DF  BU.BB [ /H /I /Border [burl@border] /Color
[burl@bordercolor]  /Action « /Subtype  /URI  /URI BU.L »
/Subtype /Link BU.B /ANN pdfmark /burl@stx null def  def
/BU.BB  burl@stx HyperBorder sub /burl@stx exch def burl@endx
HyperBorder add  /burl@endx exch def burl@boty Hyper-
Border add  /burl@boty exch def burl@topy  HyperBorder
sub /burl@topy exch def  def /BU.B  /Rect[burl@stx burl@boty
```

Jigsaw-Unintended-Bias-in-Toxicity-Classification-solution

Pengcheng Jiang

JiLin University

2021-05-20



Overview

Problem Definition

Jigsaw-Unintended-Bias-in-Toxicity-Classification-solution

given	A tagged dataset containing comments
target	detect toxic comments and minimize unintended model bias.
evaluate	ACC

Train

Data

id	target	comment_text
59848	0.000000	This is so cool. It's like, 'would you want yo...
59849	0.000000	Thank you!! This would make my life a lot less...
59852	0.000000	This is such an urgent design problem; kudos t...

Text preprocessing

Text preprocessing

- Count the total number of words contained in all texts, the maximum and minimum number of words contained in a text
- Check for missing data
- Change abbreviations to full:isn't -> is not(via dictionnary)
- clean_numbers
- Find all non alphabetic characters and clean_special_chars
- Solve the problem of misspelling words
- lower

Text preprocessing

comment_text	comment_text
This is so cool. It's like, 'would you want yo...	this is so cool it is like would you want y...
Thank you!! This would make my life a lot less...	thank you this would make my life a lot less...
This is such an urgent design problem; kudos t...	this is such an urgent design problem kudos t...
Is this something I'll be able to install on m...	is this something I will be able to install on...
haha you guys are a bunch of losers.	haha you guys are a bunch of losers

Embedding

Tokenizer

concept	What tokenizer does is actually very simple. It divides the words it sees into spaces, and then uses numbers to correspond one by one. Then we take the first num_ Words is the word with the highest frequency, others are not recognized.
Instructio	First learn the dictionary of the text, and then get the corresponding relationship between words and numbers, and then convert the text into a number string through this relationship, and then use the padding method to make up the number string to the same degree, then you can proceed to the next step : embedding

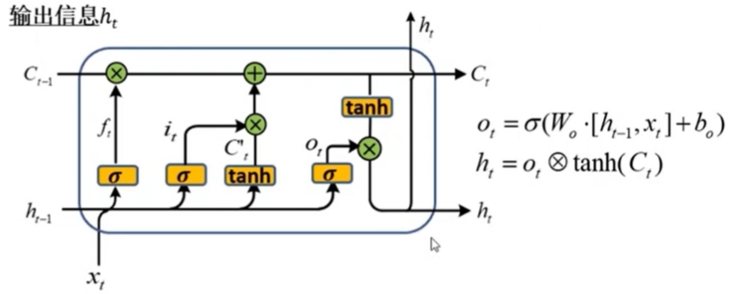
Embedding

Instructio	The embedding layer is the same as word2vec. Whether it is skip gram or cbow model, they infer each other from the context and the current, so we consider the relationship between the preceding and the following.

moudle

LSTM

Instructio	The embedding layer is the same as word2vec. Whether it is skip gram or cbow model, they infer each other from the context and the current, so we consider the relationship between the preceding and the following.
output	The embedding layer is the same as word2vec. Whether it is skip gram or cbow model, they infer each other from the context and the current, so we consider the relationship between the preceding and the following.
Dense	Map the result of LSTM to 0-1,activation='sigmoid'



$$o_t = \sigma(W_o \cdot [h_{t-1}, x_t] + b_o)$$
$$h_t = o_t \otimes \tanh(C_t)$$

Result

loss: 0.1937 - acc: 0.9401