Text Classification on Construction Description

26.04.2021

—

Madhu Pasula

Goal :

The objective of this use-case is to identify the construction code (Target) for the given construction description.

Dataset Overview :

1. Features in the Dataset : ['Construction Description', 'Construction Code'].
2. Uniques values in dataset according to each column :

Construction Description	1493
Construction Code	7

3. The shape of the train dataset is (1502, 2)
4. Number values in Target variable for every category :

```
df["Construction Code"].value_counts()
```

```
4          688
2          282
3          197
1          168
6          101
5           65
Unknown      1
Name: Construction Code, dtype: int64
```

5. Data set Full Info:

<class 'pandas.core.frame.DataFrame'>

RangeIndex: 1502 entries, 0 to 1501

Data columns (total 2 columns):

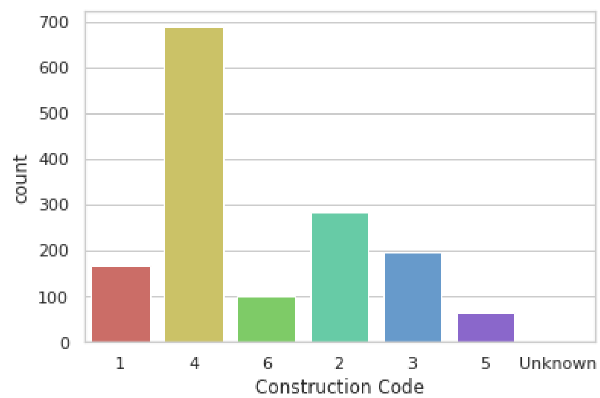
#	Column	Non-Null	Count	Dtype
---	------	--------------	-----	--------------
0	Construction Description	non-null	1502	object
1	Construction Code	non-null	1502	object

6. Null values in Target variable : 0
7. Uniques values in dataset according to each column :

Construction Description	1493
Construction Code	7

EDA :

1. Count of the values corresponding to each category :



<Figure size 432x288 with 0 Axes>

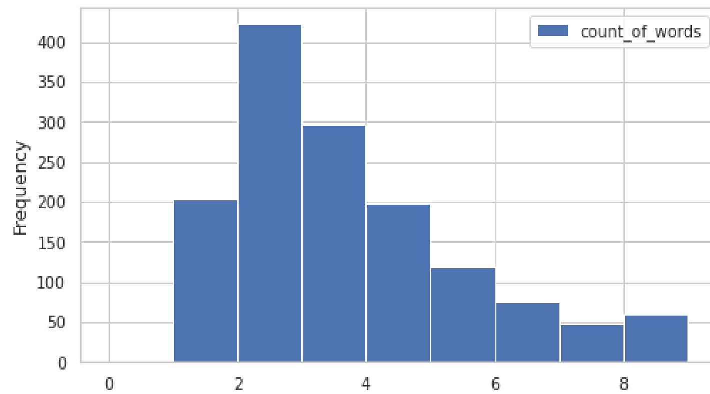
From the above plot we can understand that, the class label 4 is little daminent and has a high preference/ major role in describing the class label and it is almost 48%.

2. After removing the stop words, number, punctuations and performing the lemmatization and word tokenization :
    1. Maximum Number of letters in a Description : 456

       Minimum Number of letters in a Description : 1
    2. Maximum Number of word in a Description: 55

       Minimum Number of word in a Description: 0
3. Variance in the words for each record is being measured and found that **most of the construction description is between 2 to 5 words.**

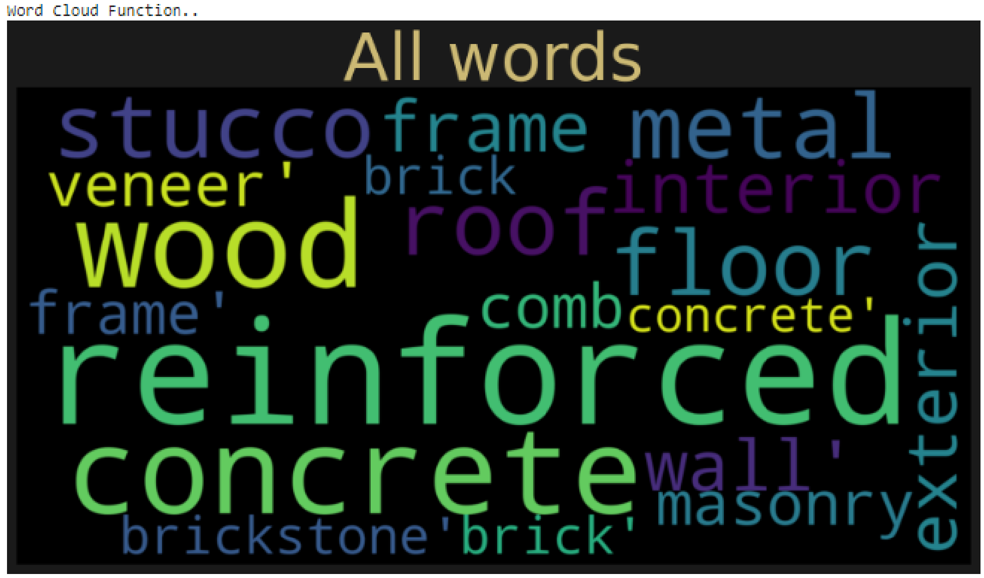
<matplotlib.axes._subplots.AxesSubplot at 0x7fb6ead42710>



From the above plot we also can conclude that the number of records which are having more number of unique words are getting flatter as the number of words are increasing. Which means there are very few descriptions that have the lengthy word spread.

4. Word Cloud : A word cloud is a simple yet powerful visual representation object for text processing, which shows the most frequent word with bigger and bolder letters, and with different colors. The smaller the size of the word the lesser it's important.

Word Cloud Function..



By the word cloud we understand that Reinforced is the most frequent word used in the construction description. Because reinforced concrete is one of the most widely used modern building materials. which means understanding this word cloud would give us the idea of business intuitions as well for this use case.

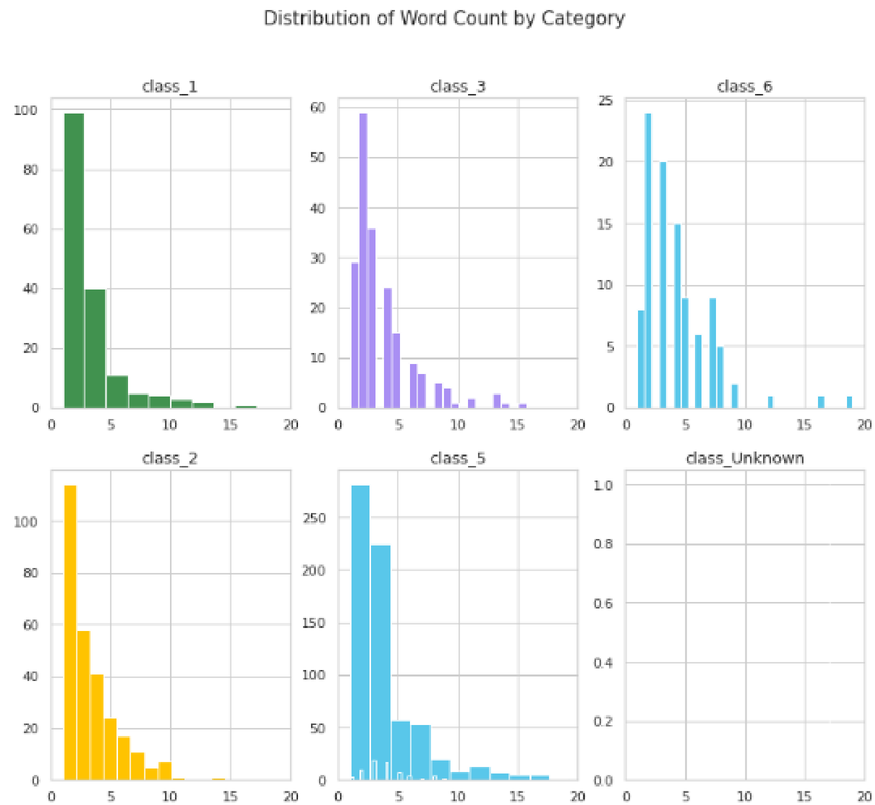
5. Top 20 words words in the Construction description :

	0	1
0	concrete	494
1	steel	425
2	frame	303
3	masonry	259
4	wood	193
5	brick	186
6	metal	184
7	block	136
8	roof	105
9	fire	104
10	and	97
11	slab	86
12	wall	85
13	exterior	83
14	deck	82
15	reinforced	82
16	stucco	79
17	resistive	76
18	veneer	56
19	tilt	48

6. Word count for the each class labels

	class_1	count	class_2	count	class_3	count	class_4	count	class_5	count	class_6	count	class_Unknown	count
0	wood	42	brick	54	steel_frame	40	concrete	154	concrete	12	fire_resistive	43	wood	42
1	frame	29	masonry	51	metal	37	steel	121	structural_steel	12	concrete	18	frame	29
2	wood_frame	28	wood	39	steel	36	masonry	71	modified	10	reinforced_concrete	16	wood_frame	28
3	stucco	18	wood_frame	32	exterior	24	brick	50	steel	8	fire	16	stucco	18
4	steel	14	concrete	30	glass	14	metal	49	fire_resistive	8	poured_concrete	10	steel	14
5	roof	14	frame	30	concrete	11	steel_frame	48	modified_fire	8	frame	9	roof	14
6	metal	9	roof	21	roof	11	concrete_block	41	resistive	8	roof	8	metal	9
7	floor	7	concrete_slab	17	noncombustible	11	block	40	mod_fire	6	fr	8	floor	7
8	concrete	7	steel	16	concrete_slab	10	reinforced_concrete	38	modified_fr	5	precast_concrete	8	concrete	7
9	veneer	6	joisted_masonry	13	metal_frame	10	masonry_steel	38	mod_fr	5	iso	7	veneer	6
10	brick	6	block	13	frame	9	roof	36	frame	4	wall	5	brick	6
11	building	6	cmu	13	panel	8	frame	33	metal	4	steel_frame	5	building	6
12	and	6	jm	10	nc	7	wall	32	mod	4	a	4	and	6
13	construction	6	and	10	masonry	7	and	26	reinforced_concrete	3	on_steel	4	construction	6
14	brick_veneer	5	concrete_block	10	aluminum	6	steel_deck	25	fire	3	poured_place	3	brick_veneer	5

7. Even the distribution of the word count for each class label also give the intuition that most the construction description is between 2-5 words irrespective of its class label.



8. After all the preprocessing, we have 859 unique tokens in over all the description corpus.

Model Architecture and Parameters Used :

```
Model: "sequential_5"
_________________________________________________________________
Layer (type)                 Output Shape              Param #
=================================================================
embedding_5 (Embedding)      (None, 60, 100)           1000000
_________________________________________________________________
spatial_dropout1d_5 (Spatial (None, 60, 100)           0
_________________________________________________________________
lstm_5 (LSTM)                (None, 60)                38640
_________________________________________________________________
dense_5 (Dense)              (None, 7)                 427
=================================================================
Total params: 1,039,067
Trainable params: 1,039,067
Non-trainable params: 0
_________________________________________________________________
None
```

Parameters :

1. Activation Function : Softmax
2. Loss = categorical_crossentropy
3. optimizer = Adam.
4. Dropouts = 20 %
5. MAX_SEQUENCE_LENGTH = 60
6. EMBEDDING_DIM = 100
7. epochs = 10
8. batch_size = 2
9. Call back saving best weights while learning rate changes at saturated loss.

Challenges:

1. Since the data set has only 1502 records with 7 classes, and most of the records on an avg have only 3 words. That's why we have very less number of unique words in the tokenizer/ word embedding. Which was creating very sparse vector If we go with the generalized embedding dimensionality and maximizing the sequence length was also not useful in learning the gradients, intuitively time consuming as well.

   Solution :

   So in order to avoid the sparsity in the word embeddings, we have chosen only 60 instead going for the big value as the maximum size of the sentence and embedding dimensionality is 100 because by our EDA we know that we have approx maximum of 55 words in descriptions. And by reducing the batch size we also tried to generalize the asymptotic test accuracy to be high for the multi class classification problem.

2. Choosing the input format for LSTM ?

   Solution :

   We have chosen a simple word embedding instead of W2V, though W2V is known to perform better. Reason behind going with tokenizer is Instead of other BOW is, we needed embedding which is less sparse and which also preserves the sequences / sentence importance i.e important in multi class classification for descriptive sentences with less parameters to be extremely efficient for converting words into corresponding dense vectors. The vector size is small and none of the indexes in the vector is actually empty.