Modern slavey Statements Detection

**A. What methodology do you propose to assess the quality of text extracted under  
column ‘TEXT’? This text is extracted from the URLs in column ‘Answer Link’. ?**

The main objective of this activity is to detect the modern slavery statements whether the company has a due diligence process in place to monitor these risks through continuous engagement with suppliers.  
After manually going through the provided data set, it is noticed that data is not ladled properly. some of the documents don't have the labels while some with other than the metric's options i.e., audits of suppliers (self- reporting), audits of suppliers (independent), on-site visits (self- reporting), on-site visits (independent), in development, no. Therefore, I selected only those documents (in function **"load\_doc"** at first conditional statement at line # 62) that are properly labelled (as per defined metric’s).

Secondly, It is also observed that there are some irrelevant / extra statements in the text (e.g., headings, company introduction). So, I defined a function **"selected\_sentences",** that takes the file as input and extract only those sentences that are talking about modern slavery or due diligence monitoring process by considering the **keywords**. These selected statements are considered for the rest of the process.

**B. Present the code of the solutions developed for this metric and interpret your results. Ensure that each section of the solution is well described and documented.**

In this sample solution, I tried to train the CNN model for the detection of modern slavery statements. I followed the following steps;

1. Loaded only those documents that are properly labelled as per metric’s using the function load\_doc. From each document only those statements/sentences extracted in which it is discussed about the modern slavery and/or monitoring process.
2. Create\_vocab() function is to pre-process (removing stop words, digits, or words with less then or equal to one char ) the documents and create vocabulary. I got the clean\_docs and vocab from this function
3. Add\_doc\_to\_vocab() is to activate/call the load\_doc() and create\_vocab() functions
4. Vocab\_select() function is to add only those words in documents text that are in the vocabulary. This will help to encode docs efficiently.
5. Lines# 152 to 162 are to encode the documents and padding the sentences.
6. Lines# 165 to 185 are to encode labels. In this encoding, each label is converted to the vector of length 6 because there are 6 metric’s i.e., audits of suppliers (self- reporting), audits of suppliers (independent), on-site visits (self- reporting), on-site visits (independent), in development, & no. If any of these metrics occurs in label that bit is turned on (1). As a result each encoded label is a list of o’s and 1’s e.g. [0,1,0,1,0,0].
7. At line # 197, I divided the training and testing data.
8. Rest of the lines are to define model, train and validate it. Model’s summary is as under

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Layer (type) Output Shape Param #

================================================================

embedding (Embedding) (None, 2080, 100) 978300

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

conv1d (Conv1D) (None, 2073, 32) 25632

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

max\_pooling1d (MaxPooling1D) (None, 1036, 32) 0

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

dropout (Dropout) (None, 1036, 32) 0

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

flatten (Flatten) (None, 33152) 0

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

dense (Dense) (None, 20) 663060

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

dense\_1 (Dense) (None, 10) 210

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

dense\_2 (Dense) (None, 6) 66

=================================================================

Total params: 1,667,268

Trainable params: 1,667,268

Non-trainable params: 0

**C. How do you assess the quality of your results? What are the challenges? What would you recommend to do to improve your initial results?**

I will assess the model on data collected from different organization. I’ll consider the following steps to improve the results.

1. I’ll train and apply a neural network model for the summarization of documents (improvement in sentences selection step)
2. I’ll try some other NN models depending on the data to make it more efficient.

**D. How would you design and deploy the Project API to align your solution with the  
WikiRate platform?**

I would like to train a NN model and deploy it as API with a user interface so that it can be used to analyse the documents.