MSA 8650 Text Analytics Project Report

Serkan Comu, Chris Lee, Jiahui Li

November 8, 2020

This project includes two parts: Messy text processing and classification, chatbot creation.

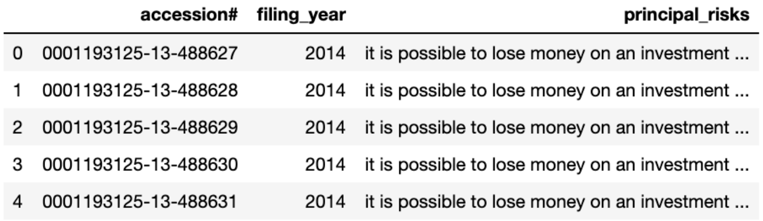
1. Messy text processing and classification

1) Motivation

We are given a large amount of messy text which we want to preprocess and then classify. In each of the messy text, there includes some kinds of investment risks that we are to classify.

2) Data

We have 5 Datasets, which totally include 81,394 documents describing investment risks in the fund investment business. Each dataset collects one year of data from 2014 to 2018.



	accession#	filing_year	principal_risks
0	0001193125-13-488627	2014	it is possible to lose money on an investment ...
1	0001193125-13-488628	2014	it is possible to lose money on an investment ...
2	0001193125-13-488629	2014	it is possible to lose money on an investment ...
3	0001193125-13-488630	2014	it is possible to lose money on an investment ...
4	0001193125-13-488631	2014	it is possible to lose money on an investment ...

Some example risk types are Market risk, interest rate risk, derivatives risk, currency risk, etc. These can be extract from the column *principal_risks*.

3) Labeling and Dataset Setup

By randomly sampling 20% observations from the datasets, we found some common patterns in the sentence that help us to divide each document into chunks. We map one label to each chunk.

Pattern 1: label is the sentence with bullet points (‘·’, ‘●’, ‘•’, ‘ª’)

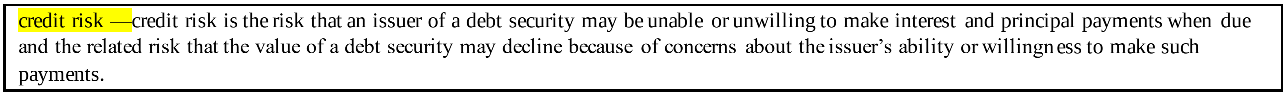
of investing in the fund as with any mutual fund, there is no guarantee that the fund will achieve its goal. the fund’s returns will vary and you could lose money on your investment in the fund. · emerging market risk. emerging market countries may have relatively unstable governments, weaker economies, and less-developed legal systems with fewer security holder rights. emerging market economies may be based on only a few industries and security issuers may be more susceptible to economic weakness and more likely to default. emerging market securities also tend to be less liquid. · foreign investment risk . foreign investing involves risks not typically associated with u.s. investments, including adverse fluctuations in foreign currency values, adverse political, social and economic developments, less liquidity, greater volatility, political instability and differing auditing and legal standards.

Pattern 2: label is the short key phrases as a sentence, for example a sentence with word ‘risk’ but the number of words is fewer than 6

interest rate risk. when interest rates rise, prices of debt securities generally decline. the fund may be subject to a greater risk of rising interest rates due to the current period of historically low rates. the longer the duration of the fund's debt securities, the more sensitive it will be to interest rate changes.(as a general rule, a 1% rise in interest rates means a 1% fall in value for every year of duration.)a low interest rate environment may prevent the fund from providing a positive yield or paying fund expenses out of current income

Pattern 3: label as ‘XXX risk:’ or ‘XXX risk – ’. The sentences after ‘:’ or ‘–‘is the explanation of the risk

equity market risk: even a long-term investment approach cannot guarantee a profit. economic, market, political, and issuer-specific conditions an
d events will cause the value of equity securities, and the portfolio or underlying fund that owns them, to rise or fall. stock markets tend to move i
n cycles, with periods of rising prices and periods of falling prices.



credit risk —credit risk is the risk that an issuer of a debt security may be unable or unwilling to make interest and principal payments when due and the related risk that the value of a debt security may decline because of concerns about the issuer’s ability or willingn ess to make such payments.

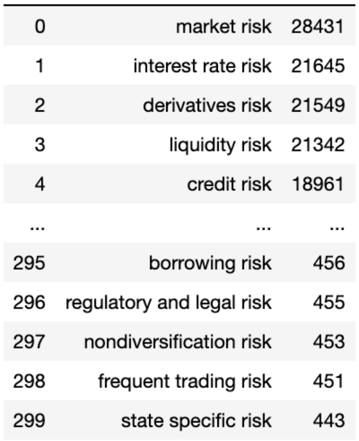
After finding the patterns, we followed these main steps to construct our dataset.

Step 1: Divide each document into multiple blocks by the observed patterns (‘●’, ‘^{a}’, short key phrases as a sentence, etc.) and extract the labels

Step 2: Cleaning the labels (remove numbers, punctuation, single-letter word, long labels with bunch of words)

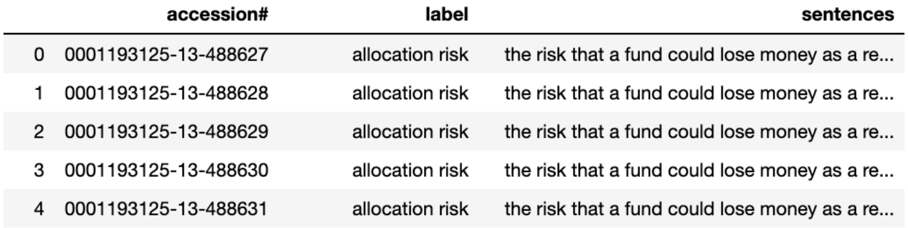
e.g. ● regulatory risk à regulatory risk

Step 3: Removed the uncommon labels and keep the top 300 labels to focus on. The reason to do this is because some labels only have few observations. Below shows the risk type we keep and the total number of observations for each risk type.



0	market risk	28431
1	interest rate risk	21645
2	derivatives risk	21549
3	liquidity risk	21342
4	credit risk	18961
...	...	...
295	borrowing risk	456
296	regulatory and legal risk	455
297	nondiversification risk	453
298	frequent trading risk	451
299	state specific risk	443

After all the initial preprocess, we created a dataset with 598,973 rows. Each row represents a block from one document with a defined label.

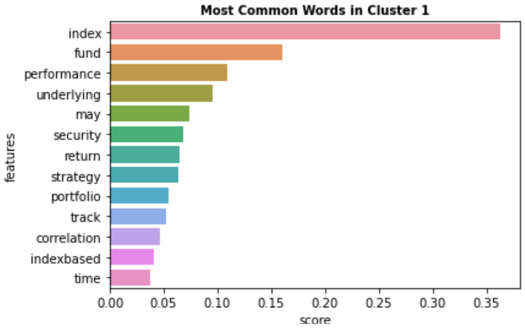


	accession#	label	sentences
0	0001193125-13-488627	allocation risk	the risk that a fund could lose money as a re...
1	0001193125-13-488628	allocation risk	the risk that a fund could lose money as a re...
2	0001193125-13-488629	allocation risk	the risk that a fund could lose money as a re...
3	0001193125-13-488630	allocation risk	the risk that a fund could lose money as a re...
4	0001193125-13-488631	allocation risk	the risk that a fund could lose money as a re...

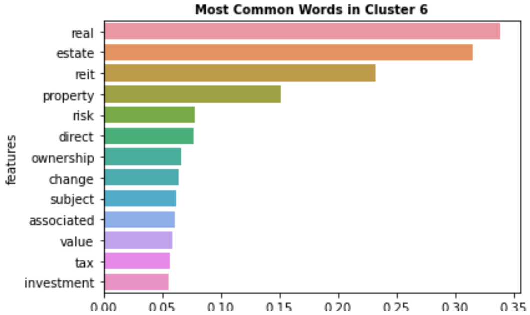
4) Model Approach

In this project, since we had 300 unique labels, we decided to cluster them into more manageable chunks. Unfortunately, since we had too much data, we were unable to make the TF-IDF vectorizer to work without 200 gigabytes of ram. Therefore, we decided to use the sampling function of pandas groupby and took 400 samples of every label we had. This gave us 120000 rows of data which was more manageable. After we sampled down our data, we used TF-IDF to vectorize our sentences after which we ran a k-means clustering algorithm on our training data. Using the elbow method, we saw that 15 groups was suitable for this dataset.

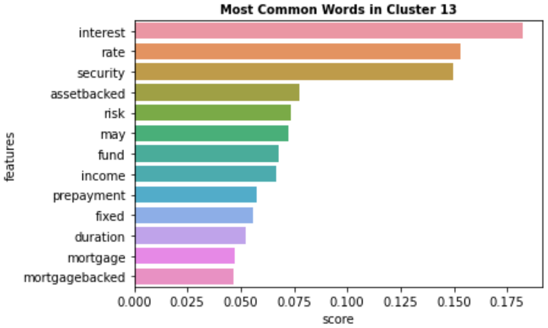
Here are a couple interesting clusters with their 5 most common labels:



index based investing
index fund risk
index performance risk
passive strategy index risk
index related risk



real estate risk
real estate securities risk
real estate industry risk
reits risk
real estate investment risk



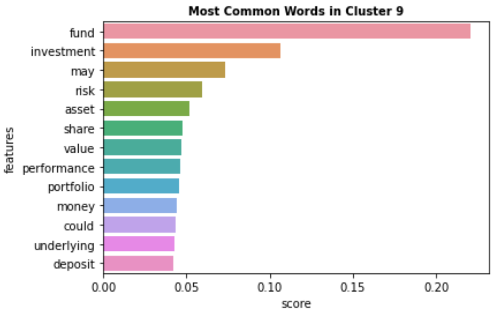
asset backed and mortgage backed securities risk
asset backed securities
fixed income securities
prepayment or call risk
duration risk

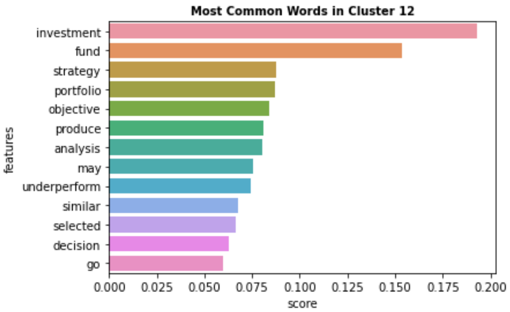
We then got rid of all sentences that were longer than 4000 words, after which we got dummies for our labels and then put our data through our model.

We started by initializing a sequential model followed by an embedding layer. Then we used a LSTM layer with 30 neurons, and a dense layer with 30 neurons followed by a 15-neuron output layer with one neuron corresponding to every cluster. We used adam as the optimizer and binary_crossentropy loss. We use the included accuracy metric for our results, and we used a batch size of 32 to train our model.

4) Model Result

Our model gave us a 94 percent accuracy so we can say that the model was good at spotting data that came from our sample data. Since we used clustering however, our model is only as good as our clustering. The results are accurate to the clustering however some of the labels was spread out over the clusters leading us to think clustering with all 400 samples of each label being taken as one document might be better for a grouping approach in the future. We also wanted all clusters to be distinct however some were very similar to each other like 9 and 12.





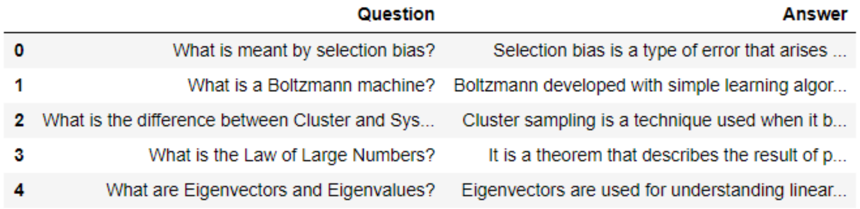
2. Chatbot Creation

1) Motivation

We want to build a chatbot that is related to data science and machine learning questions and answers. We want the chatbot to be able to generate new answers to questions asked.

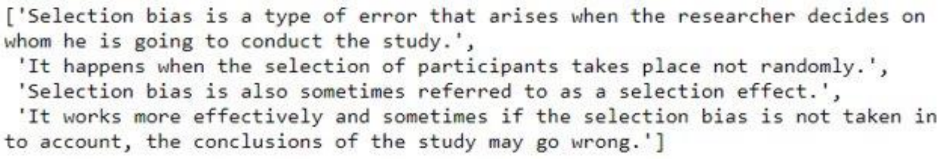
2) Data

For our data, we manually took Questions and Answers from six different sources and compiled them together. We compiled a little over 300 observations from these sources. The data consists of only two columns, Questions and Answers.



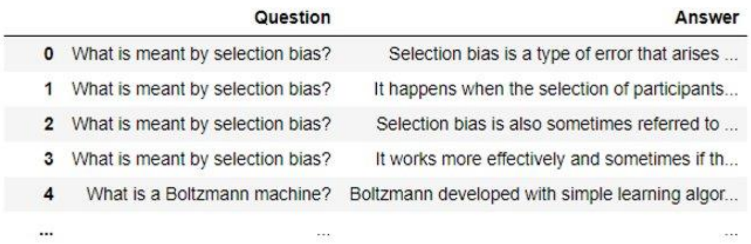
	Question	Answer
0	What is meant by selection bias?	Selection bias is a type of error that arises ...
1	What is a Boltzmann machine?	Boltzmann developed with simple learning algor...
2	What is the difference between Cluster and Sys...	Cluster sampling is a technique used when it b...
3	What is the Law of Large Numbers?	It is a theorem that describes the result of p...
4	What are Eigenvectors and Eigenvalues?	Eigenvectors are used for understanding linear...

Because some of the Answers to Questions are just astronomically large, we want to split them based on sentences or blocks which in the end will help us get more observations. The following shows the first observation's answers being split on sentences.



```
['Selection bias is a type of error that arises when the researcher decides on
whom he is going to conduct the study.',
 'It happens when the selection of participants takes place not randomly.',
 'Selection bias is also sometimes referred to as a selection effect.',
 'It works more effectively and sometimes if the selection bias is not taken in
to account, the conclusions of the study may go wrong.']
```

Here, we split the answers for the first question, and as we saw before, there were 4 sentences in the Answer for the question "What is meant by selection bias?" With this one entry from the raw data, we converted it to 4 observations/entries.



	Question	Answer
0	What is meant by selection bias?	Selection bias is a type of error that arises ...
1	What is meant by selection bias?	It happens when the selection of participants...
2	What is meant by selection bias?	Selection bias is also sometimes referred to ...
3	What is meant by selection bias?	It works more effectively and sometimes if th...
4	What is a Boltzmann machine?	Boltzmann developed with simple learning algor...
...	...	...

After the splitting of all our data's answers and mapping by question, we generated about a total of 1400 observations.

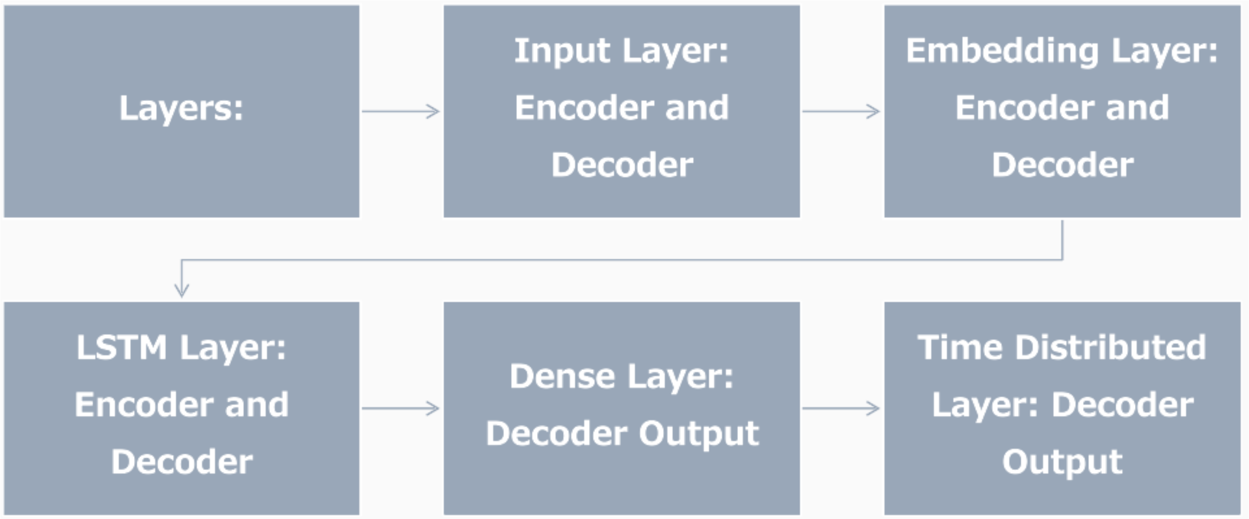
3) Method

For the creation of a Generative Chatbot for data science interviews Questions and Answers, we used the RNN type many to many, sequence to sequence model. This approach involves two Recurrent Neural Networks (RNNS). One to encode the source sequence, called the encoder, in our case, the Questions. The second one to decode the encoded source sequence into the target sequence, called the decoder, in our case, the Answers.

In our first attempt, we did regular preprocessing of our data -- Removing punctuations, tokenizing every answer, removing all the stop words, stemming all the words to their root, and then joining them back to a new preprocessed answer sentence. After preprocessing the data, we defined a vocabulary size, which is the number of unique words combined in both the encoders and decoders. We use the vocabulary size to help make our word to index and index to word to make our currently string valued Q&A's into sequences. After converting all the Encoders and Decoders, we padded them with the max length of the encoder and max length of the decoder so that all the encoder and decoder sequences are of equal length.

For the Embedding Matrix and Embedding Layer, we used a Pre-Trained matrix from GloVe, a global vector for word representation, which is an unsupervised learning algorithm for obtaining vector representations for words.

4) Model


Layers:
Input Layer: Encoder and Decoder
Embedding Layer: Encoder and Decoder
LSTM Layer: Encoder and Decoder
Dense Layer: Decoder Output
Time Distributed Layer: Decoder Output

Of course, we have our input layer, The Encoder and Decoders which inputs out sample, with the max sequence length, and an output wit max output length.

Following that we have the embedding layer of the encoder and decoders, which embeds our encoders and decoders, based on our vocab size, the number of unique words, and our sequence lengths, to output a new layer now of 3-D, the number samples, sequence length, and embedding dimensions.

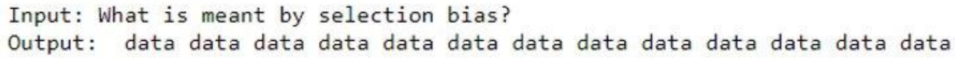
Following that is the LSTM layer of the encoders and decoders, where LSTM can add or remove information. Like as it is processing, whether to return to the last state along with the output; *or* to return a sequence; whether the last output of the output sequence or a complete sequence is returned.

Following is the dense layer and the time distributed layer. Both these layers are applied to the decoder. The time distributed layer allows us to apply a layer to every temporal slice of an input.

With these layers, we have trained our model with multiple different parameter values. Varying activation functions, batch sizes, shuffling, epochs, and optimizers.

5) Results

While varying the parameters, our best model produced an accuracy of ~84%. Although with a decent accuracy rate, when testing the results, our output was not good. It had repeated values with no sentiment generated solutions. Our outputs looked like this.



```
Input: What is meant by selection bias?
Output:  data data data data data data data data data data data data data
```

When we sought out the possibilities for this, we realized we should not preprocess the data too much because we lose a lot of sentiment when training from the encoders to decoders. In normal preprocessing, removal of stop words and stemming and such is a staple, but not in this case. In our following attempt for running the model without much preprocessing, we received an extremely low accuracy of about 20%. We believe that this occurrence is primarily due to our lack of data. A little over a thousand entries is not nearly enough to produce a decent generative chatbot. For future endeavors, we would adopt more data and when preprocessing, we would at most make every word uniformly lowercased and formal; remove convert contractions into their full meaning for more semantic training of the data.

If we had more time and data, we wanted to implement both a conditional retrieval based and generative chatbot, where if the input has a high matching value to one of the training, it will output the best solution that was matched to that input, and if not, it will generate an answer with generative chatbot.