Name: Ananth (Anantharam) Subramanian

Mid-Term Examination- Natural Language Processing

# Short Essay Responses

1. Select one career or industry that makes use of applied NLP.
   1. Explain generally how that field or career utilizes NLP.

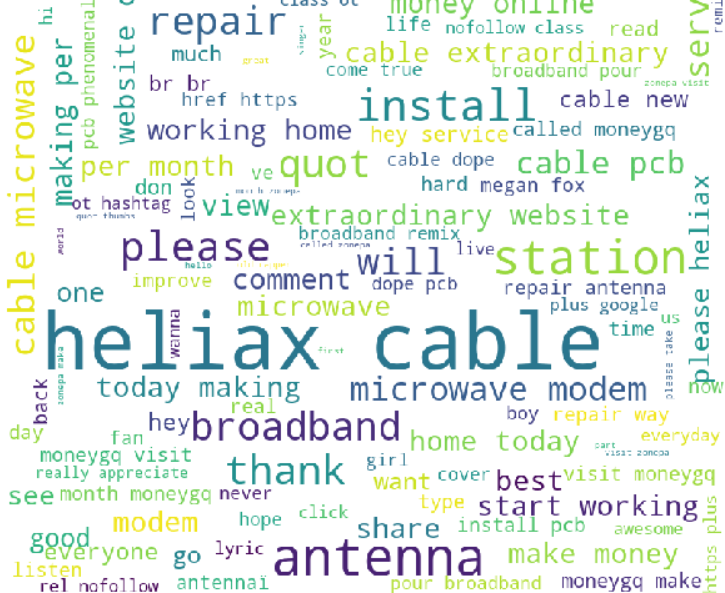
The chosen industry is Telecom that manufactures mobility and connectivity devices. One example of mobility device is Base Station Antennas and connectivity device is modem.

The industry has its own e-commerce website for selling the products and a helpdesk to route the customer requirements to appropriate departments. The information passed on by the customers are collected in a database for further analysis and to take decisions.

Similarly, speech recognition tools are used to convert the voice conversations to text and further parsed to create meaningful information.

* 1. Explain at least some methods of NLP that are very likely to be used in the career or industry you selected.

i) Natural Language Processing methods like word cloud are used to understand the frequently discussed or ordered product (for example: antenna, broadband, modem, cable, etc.). This will further assist the Product Owners to understand the quality issues and improvements needed. A sample word cloud for the telecom industry I have chosen could look like this:



ii) Most of the industries not limiting to Telecom Industry use an ERP system like SAP. Business Intelligence is one of the modules within the ERP system and in order to provide the top management and departments like sales and marketing with the right information (example: most bought products, top quality issues reported, competitor product information and corresponding sales figures) reports are generated on the data extracted from the ERP system. Natural processing methods like semantic analysis to understand the relationship between the words from the data extracted and provide appropriate information to Business Intelligence.

Another area of Natural Language Processing that is being utilized is translation ability (ex. English to German and vice versa). The products are being sold in multiple countries and in order to generate product information in different languages, the data extracted are translated to be put on flyers and catalogs. [My understanding is that stacked LSTM can be used for translation. <http://www.manythings.org/anki/deu-eng.zip> provides sample data set for English to German translation. Both shorter and longer phrases are covered]

* 1. Give at least one specific example of a use case for NLP within the chosen field and explain how the problem or situation is (or could be) improved by applying NLP.

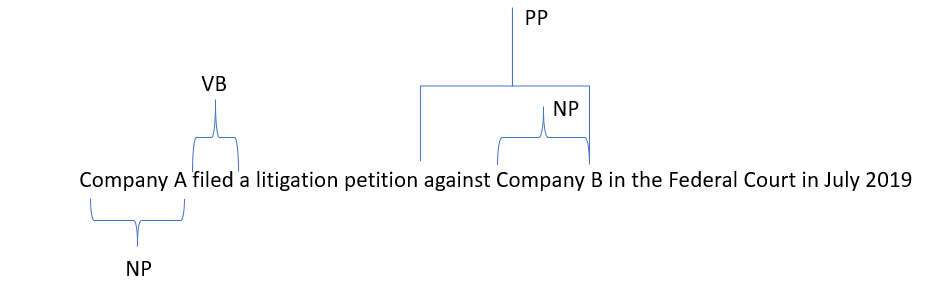
If someone is interested in tracking business news, one of the news items was related to a litigation filed in the Federal Court. It was filed by Company A ( a famous telecom company in the Dallas area) against Company B ( a Germany based telecom company and operating out of New Jersey) for stealing its trade secrets. The legal department of Company A can take the help of Natural Language Processing word parsing on email messages exchanged for several years and do a comparison of the frequently used words with its patent applications and draw a similarity (GloVe, SpaCy can come to its rescue) and compare with the product features being sold by Company B. This will help in accelerating the process as the legal department does not have to spend several months analyzing the emails.

1. Choose one of the “trade-offs” in NLP that was covered in the materials for this course.
   1. Explain the trade-off in general terms. Define the two choices.

Trade-off is choosing a method from available methods based on the scenario. There are few trade-offs that are available in NLP and they are “Shallow vs Deep”. The others include “Statistical vs symbolic”, “Feature Engineering vs Feature Learning”, “Top-down vs Bottom-up” and “Transparent vs Opaque”.

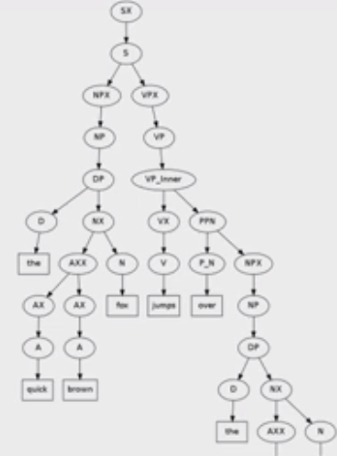
**Shallow:** This method covers parsing at a high-level rather than going deep into the analysis. For example let us say we have the statement:

“Company A filed a litigation petition against Company B in the Federal Court in July 2019”



The syntactic analysis include using a shallow parser leaving out multi phrases as single unit and returning the part-of-speech tag. For the sematic analysis, it will return who (Company A) is filing what (litigation petition) against who (Company B) where (Federal Court) and when (July 2019). The Stanford Shallow Sematic analyzer helps with this analysis.

**Deep:** This is just opposite of Shallow and it does deep parsing ie. it parses each token. Here is a pictorial representation of a deep parsing.



Similarly, the semantic analysis using Deep parsing involves extracting the meaning of each and every word.

* 1. Explain the benefits and weaknesses of each side of the trade-off. Include at least one benefit and one weakness of each.

As described earlier, the methodology that needs to be adopted depends on the scenario. Suppose we need to scrap some high-level information from different set of documents not worrying about the background, shallow parsing will help. On the other hand, if we need to do a deep syntactic and semantic analysis, deep parsing is the way to go. Deep parsing will help in getting more information and helps in accurate prediction.

**Shallow:**

**Benefit:** Not machine intensive. If the requirement is such that information is needed at a higher-level , Shallow parsing helps achieve that very quickly and efficiently

**Weakness:** We might miss some important information because of the shallowness. If higher accuracy needed in classification, this is not the correct method.

**Deep:**

**Benefit:** Provides detailed information on the documents after doing a deep parsing. This helps in taking meaningful decision as lot of information is available at one’s disposal. Gives greater accuracy with respect to classification.

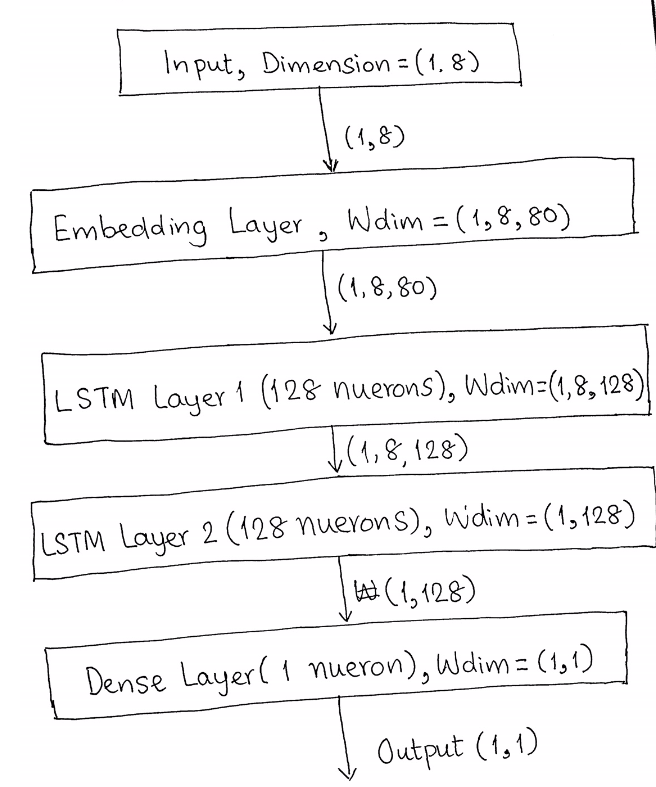
**Weakness:** Machine intensive and takes lot of GPU/CPU for turning the output. It also takes several weeks if not months to provide the information needed.

c. Describe a work-situation that would make one of the choices in the trade-off much better, in terms of practical outcomes for you and your stakeholders on a project.

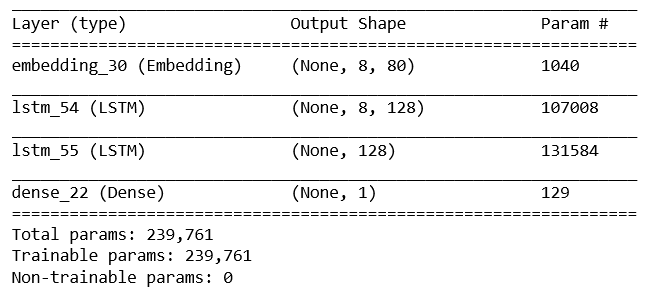
With reference to the earlier news item regarding a Telecom company suing another Telecom company for stealing its trade secrets, one of the trade-offs helps to provide the legal department of Company A the information it needs to secure win in the Federal Court. We had talked about scanning through email messages of several years to obtain the details that were exchanged regarding the product and to find the corresponding matches the Intellectual Property documents. Deep parsing will come to rescue as it helps to provide a detailed information especially the Semantic analysis. It needs special skill to compose Intellectual Property documents and the lawyers will be able to quickly find the similarity of the words between the email messages and the Intellectual Property documents and this will help in nailing down if Company B has indulged in wrongdoing.

# II NLP Networks:

1. Label each block and step by input/sequence step. Compute the dimensions of the weight matrix for the output All inputs must be labeled by dimension. Include your original word vector as input



**Model Summary:**



1. Write the initial vector form of the input sequence

Input Vector after padding:



1. Find the average Glove Word Vector you the input sequence (Spacy uses Glove vectors!)

The following procedure was used to find the average glove word vector for the input sequence ( I have used Spacy to get the necessary information):

# Imports

from scipy.spatial import distance

import spacy

# Load the spacy vocabulary

nlp = spacy.load("en\_core\_web\_lg")

input\_word1 = "The"

input\_word2 = "car"

input\_word3 = "jumped"

input\_word4 = "over"

input\_word5 = "the"

input\_word6 = "fence"

w\_1 = np.array([nlp.vocab[input\_word1].vector])

w\_2 = np.array([nlp.vocab[input\_word2].vector])

w\_3 = np.array([nlp.vocab[input\_word3].vector])

w\_4 = np.array([nlp.vocab[input\_word4].vector])

w\_5 = np.array([nlp.vocab[input\_word5].vector])

w\_6 = np.array([nlp.vocab[input\_word6].vector])

q = w\_1 + w\_2 + w\_3 + w\_4 + w\_5 + w\_6

p = q/6

There is a better way to do it but in order to calculate the output word provided the input vector, the above procedure helped.

[average\_word\_vector = np.mean(word\_vector\_list, axis=0) can also be used]

The average vector is stored in a text file.



1. Find the nearest word to answer #3

Nearest word is “Car”

The following procedure was used:

# Format the vocabulary for use in the distance function

ids = [x for x in nlp.vocab.vectors.keys()]

vectors = [nlp.vocab.vectors[x] for x in ids]

vectors = np.array(vectors)

# \*\*\* Find the closest word below \*\*\*

closest\_index = distance.cdist(p, vectors).argmin()

word\_id = ids[closest\_index]

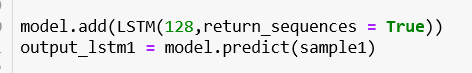
output\_word = nlp.vocab[word\_id].text

[I referred this thread in stackoverflow to arrive at the answer:

<https://stackoverflow.com/questions/54717449/mapping-word-vector-to-the-most-similar-closest-word-using-spacy>]

1. What is the difference between the W matrix the first LSTM sequence at time/sequence 0 and at time/sequence 5

The output from first LSTM layer is captured using:



This will have outputs from 0 through 5 sequences

The output at seq 0 is captured using:



The output at seq 5 is capture using:

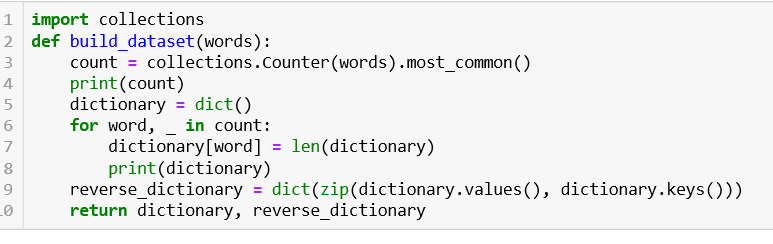


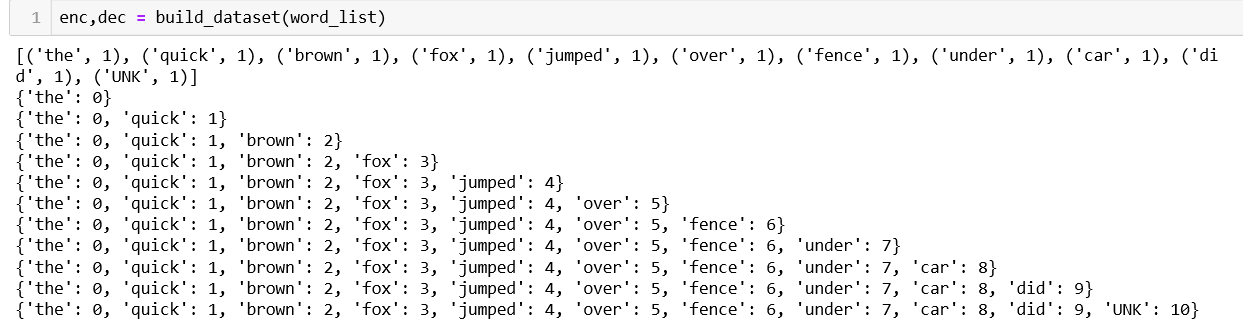
The difference is stored in the attached file:



1. What is missing in the above code—something important is not determined and based on that, there are some minor adjustments or additions that need to be made Make a logical determination of what that missing piece of info should be based on the info given here and what additions or adjustments are necessary.

* The top words have not been defined. I first created a dictionary (thanks to your lecture on Spam email detection) and used it for assigning the index to the top words.





* Even though the ask is not for building a neural network for classification, doing this will help in extending to several sentences.
* Indexes are assigned to the input sequences. This is also missing in the given code.



Though this is not needed for answering the questions, this will help in extending the network to classify sentences as “statement” or “question”.

* I have used the concept above to analyze the sentence “The car jumped over the fence”.

In order to arrive at the output weights, I had to include lines and comment out lines as specified below.

The top words numbers used is 13 as I have included for padding and start, so that index of the first word starts from 2.



The line output\_lstm1 = model.predict(sample1) provides the weight matrix for LSTM layer 1