LANGUAGE TRANSLATION

(ENGLISH-SPANISH)

OBJECTIVE

- The objective is the machine translation from English to Spanish limited by the scope of dataset.
- A system will be created which translates the words or phrase between any two languages and provides us with the broader context of the word or phrase.

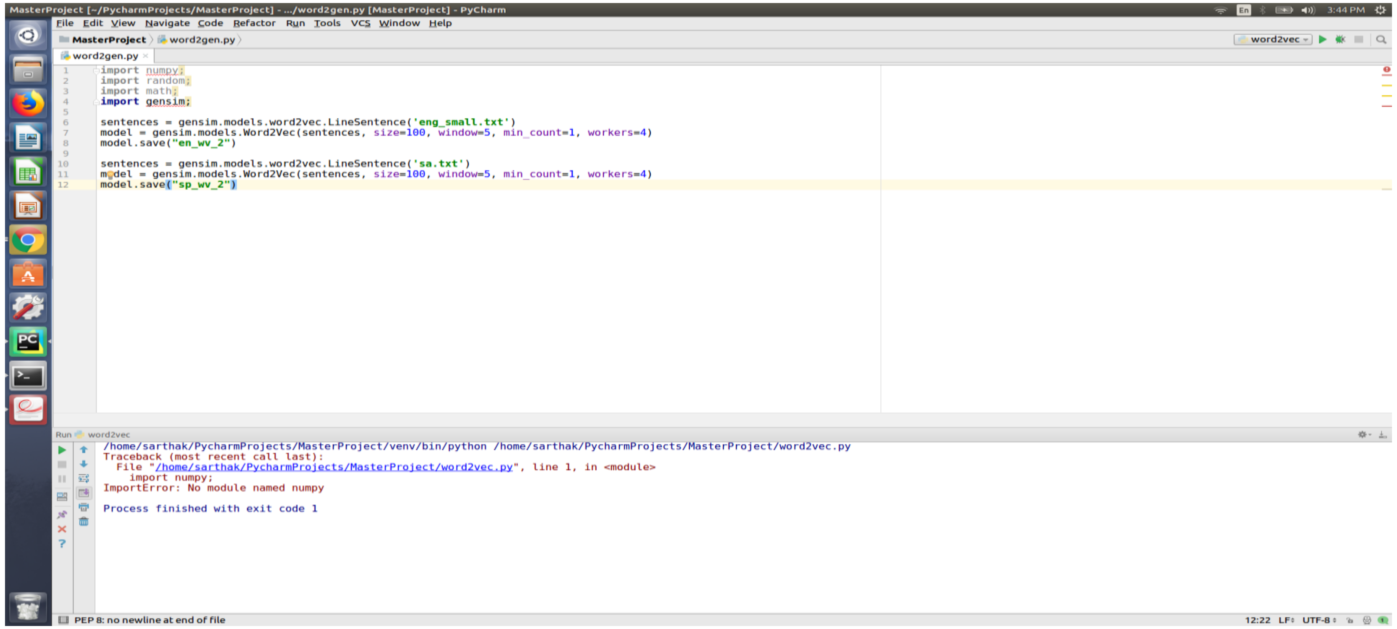
METHODOLOGIES

- **DATA COLLECTION**

  I started the project by selecting the training data, I chose data from United Nations proceedings available at Linguistic Data Consortium (LDC).All the data in English was line by line converted to Spanish in the training data set.

- https://conferences.unite.un.org/UNCorpus/

- **IMPLEMENTING Word2Vec(completed)**
- Word2vec was used and trained on both English and Spanish training data and the model was saved.
- Gensim library was used to implement Word2vec.
- https://radimrehurek.com/gensim/tutorial.html


MasterProject [~/PycharmProjects/MasterProject] - .../word2gen.py [MasterProject] - PyCharm
File Edit View Navigate Code Refactor Run Tools VCS Window Help
MasterProject word2gen.py
word2gen.py
word2vec
Run word2vec
PEP 8: no newline at end of file
12:22 LF UTF-8

```python
import numpy;
import random;
import math;
import gensim;

sentences = gensim.models.word2vec.LineSentence('eng_small.txt')
model = gensim.models.Word2Vec(sentences, size=100, window=5, min_count=1, workers=4)
model.save("en_wv_2")

sentences = gensim.models.word2vec.LineSentence('sa.txt')
model = gensim.models.Word2Vec(sentences, size=100, window=5, min_count=1, workers=4)
model.save("sp_wv_2")
```

```
/home/sarthak/PycharmProjects/MasterProject/venv/bin/python /home/sarthak/PycharmProjects/MasterProject/word2vec.py
Traceback (most recent call last):
  File "/home/sarthak/PycharmProjects/MasterProject/word2vec.py", line 1, in <module>
    import numpy;
ImportError: No module named numpy

Process finished with exit code 1
```

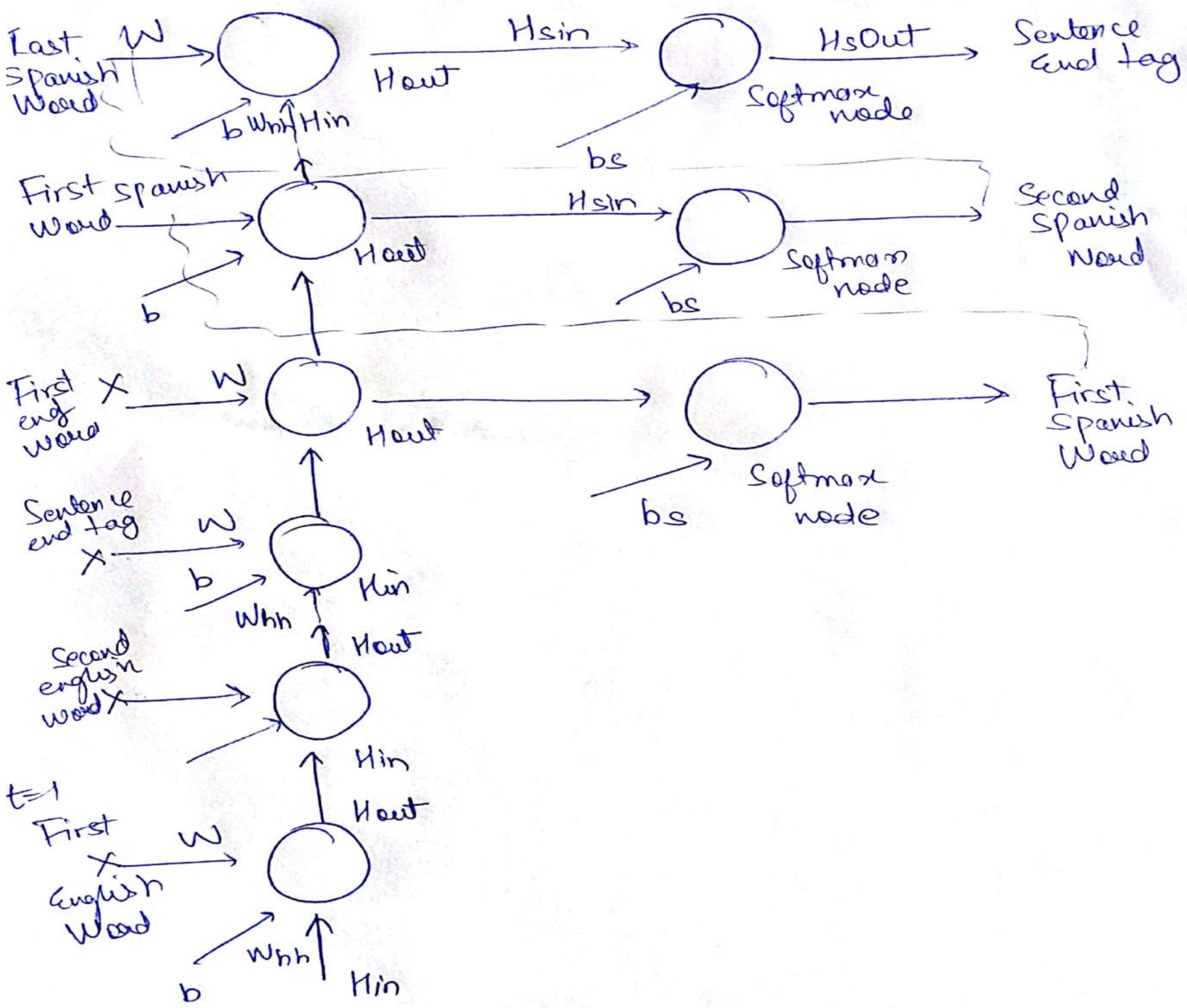
- The word vector was generated and sent to RNN for translation.

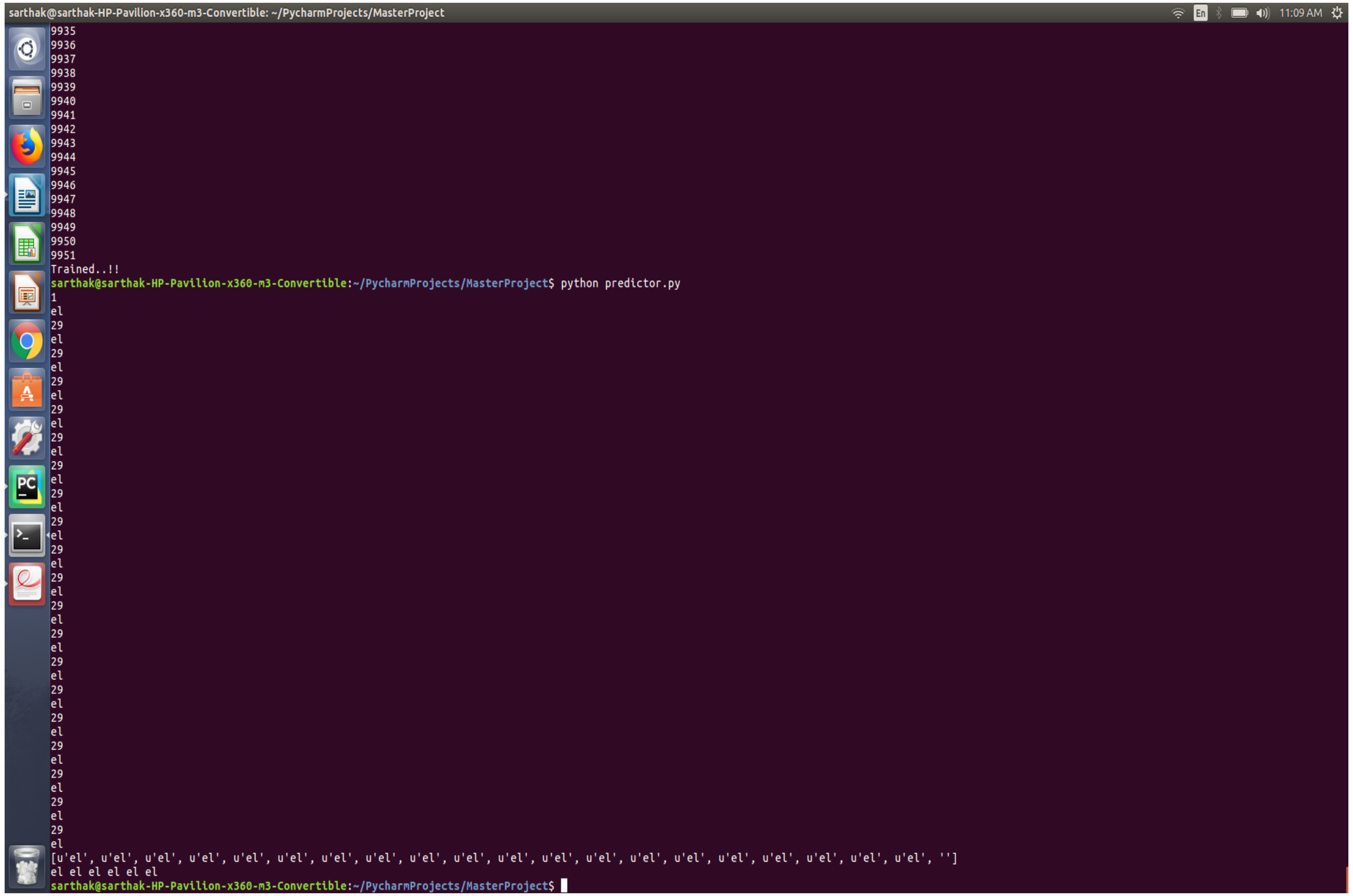
RNN(Recurrent neural network)(completed)

- A single neuron with 3 inputs (including the memory from the previous node, the English word and the Bias) and weight vectors unfolding through time.
- For each sentence, RNN is trained and each English word is being given as input in consecutive time periods

- After all the English words in as sentence are processed through, I pass “End of the statement” tag to RNN Neuron as input.
- This EOS will change the inputs from English words to the prediction words.
- Now after EOS, I use the concepts in LSTM and supply the first word as input to the next neuron to give it a little extra push for the prediction of the corresponding first Spanish word.
- SoftMax is applied on the output of this neuron to get the index of the predicted word in the vocabulary.

RNN Network


Last Spanish Word
W
b Whh Hin
Hout
Hsin
bs
HsOut
Softmax node
Sentence End tag
First spanish Word
b
Hout
Hsin
bs
Softmax node
Second Spanish Word
First eng word X
W
Hout
bs
Softmax node
First Spanish Word
Sentence end tag X
W
b
Whh
Hin
Hout
Second english word X
Hin
Hout
t=1
First English Word X
W
b
Whh
Hin



sarthak@sarthak-HP-Pavilion-x360-m3-Convertible: ~/PycharmProjects/MasterProject
En
11:09 AM

```
9935
9936
9937
9938
9939
9940
9941
9942
9943
9944
9945
9946
9947
9948
9949
9950
9951
Trained..!!
sarthak@sarthak-HP-Pavilion-x360-m3-Convertible:~/PycharmProjects/MasterProject$ python predictor.py
1
el
29
el
29
el
29
el
29
el
29
el
29
el
29
el
29
el
29
el
29
el
29
el
29
el
29
el
29
el
29
el
29
el
29
el
29
el
29
el
[u'el', u'el', u'el', u'el', u'el', u'el', u'el', u'el', u'el', u'el', u'el', u'el', u'el', u'el', u'el', u'el', u'el', u'el', u'el', u'el', '']
el el el el el el
sarthak@sarthak-HP-Pavilion-x360-m3-Convertible:~/PycharmProjects/MasterProject$
```

BACK-PROPAGATION(still working on it)

- But after this, in training phase I will compare the predicted word vector and with the original target Spanish word vector to calculate the Error for back propagation and pass target Spanish word vector to the next neuron as input.
- While in the testing/prediction phase, we pass the predicted word vector to the next neuron as input.

TESTING AND COMPARING(to start)

PROBLEMS FACED AND FACING

- Unavailability of Large Data and so the model is not getting trained properly. Need more data .
- Running the RNN for more epochs will also result in significant improvement. Right now I am working with epochs number 15 since my data is not much and to process this data also it took me 12 hrs to train.
- Concept of Softmax was tricky and took time to run.
- Initially gensim was not working properly in windows and so I am doing it in UBUNTU.
- Many packages were installed to import various libraries like cpickle,gzip as the system was finding compatible issues.
- Not able to install numpy ,pandas on pycharm ,running the code on the terminal.

SOME REFERENCES

- https://pdfs.semanticscholar.org/8b9c/a3c24d89851f276a7b13413bd047b8577999.pdf
- http://cs231n.github.io/linear-classify/
- http://colah.github.io/posts/2015-08-Understanding-LSTMs/
- https://iamtrask.github.io/2015/11/15/anyone-can-code-lstm/
- https://arxiv.org/pdf/1409.0473.pdf
- http://ai.dinfo.unifi.it/paolo/ps/tnn-94-gradient.pdf
- https://github.com/lesley2958/word2vec#12-autoencoders
- Some reference was taken from COURSERA MACHINE LEARNING CERTIFICATION.