Hacettepe University

Statistical Natural Language Processing

Assignment-3 :

A Neural Language Model with a Feedforward Network Using DyNet

---

Author:
Efsun Sezer
N16244397

Instructor:
Assoc. Prof. Dr. Burcu CAN

May 16, 2019

1. Word Level Neural Language Model

The aim of the language modeling is to predict the next word or character in sequences. It is a key point of NLP and there are so many applications that require language modelling such as machine translation, text generation etc. Language model can be operated at character level, word level or even sentence level. In this part, a word-level neural language model is examined.

Steps for building word-level language model are as follows:

- As a starting point, data is prepared for the neural network model. For that purpose, data is loaded and processed. Processing step involves punctuation removing, converting text case to lover case. After that, a vocabulary consisting of unique words seen with a specific frequency in the given corpus is created.
- Input and output word pairs are generated and mapped to integers using the vocabulary. Integers values are then converted to one-hot vectors.
- Generated samples are used to train a neural language model which has one hidden layer. Computational graph for that model is given in Figure 1.

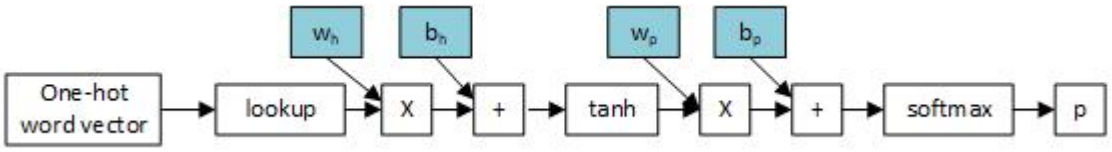

Figure 1: A computation graph for a bi-gram feed-forward neural language model

1.1. Results

In this part, example sentences that are produced by word-level neural network language are listed below (start word is marked with bold):

1. **washington** is so many websites many websites many websites
2. **republican** politicians talk about it is so many
3. **ambassador** I have to court and they talk about it so many
4. **evangelicals** I have to do anything about it is so many
5. **disgraceful** it is very hard for politics run for politics

2. Character Level Neural Language Model

Pipeline for building character level neural language model is similar to word level model. In order to train the language model, we need to convert characters into meaningful vectors. This is performed by building a vocabulary that contains all the unique letters in the given corpus. A vocabulary is used to encode the input and the output characters as integers. One-hot vectors of the inputs fed into the model for training and testing.

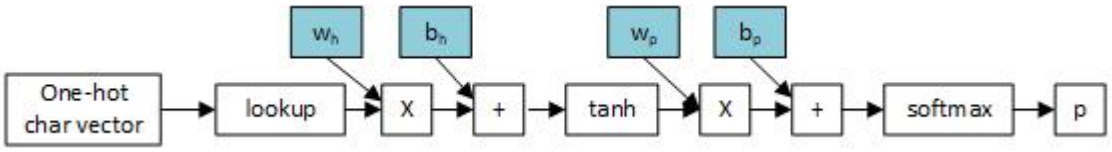

Figure 2: A computation graph for a character-level feed-forward neural language model

2.1. Results

Char-level neural language model tries to predict a character based on the previous character. Results are shown below (start characters is marked with bold):

1. **m**and the the the the the the the th
2. **,** the the the the the the t h e the the the the the t

3. **5** the the the the the the t h e the the the the the t

4. **u**s t t t t t t t t t t t t t t t t t t t t t t t t t t

5. **q**uthe the the the the the the the the the the the

Unfortunately it is very hard to draw clear inference from these result. Maybe trying larger char n-grams, adding extra hidden layers or using different deep learning architectures such as RNN, LSTM, GRU may improve the results.

3. Conclusion

Classical N-gram models have one discrete representation for each word. In these models, usage of larger vocabularies of words increases the dimensionality of the vectors and results in more sparse representations.

Neural Language Models (NLM) solves the data sparsity problem, by representing words as vectors and using them as inputs to a NLM. Different from classical methods, NLM associates each word in the vocabulary with a distributed feature vector.

The distributed representation is highly effective at capturing relationship between words because similar words have similar word embeddings and these vectors are mapped to similar positions in vector space.

Neural Language Models (NLM) may utilize different information such as word level, character level or morpheme level.

Word-level language models may produce high quality of word vectors for frequent words but their performance deteriorates as the frequency of the words decreases.

To overcome this sparsity issue, researchers introduced character level language models. Character level model shows promising results in the case of rare words. However their main drawback is that they are bad at dealing with the words that are semantically similar but orthographically different.

References

[1] How to create a poet / writer using Deep Learning?
`https://www.analyticsvidhya.com/blog/2018/03/text-generation-using-python-nlp/`

[2] Text Vectorization and Transformation Pipelines
https://www.oreilly.com/library/view/applied-text-analysis/9781491963036/ch04.html

[3] Bengio et al. A Neural Probabilistic Language Model, Journal of Machine Learning Research, 2003.