WAW – Machine Learning 6

Tutorial: NLP with Python

Sivasurya Santhanam

Intelligent software systems

Institute for Software Technology (SC-IVS)

27.10.2020

Agenda

- 9:30 - 10:30 : Part I (Pre-processing of text)
- 10:30 - 10:45: Break
- 10:45 - 11:45: Part II (Sentiment analysis)
- 11:45 - 12:00: Break
- 12:00 - 12:30: Part III + Q & A

Why is NLP hard?

- Representation of semantic meanings and contexts
- Syntax, Semantics, pragmatics
- Humans also apply sarcasm now and then
- Accents and dialects (Speech recognition)

Part - I

Pre- processing

Words & representations – Bag of words

Example:

1. I read a book about book reading

	a	about	book	I	read	reading
Sentence-1	1	1	2	1	1	1

Words & representations – Term-Document matrix

Example:

1. This sample is a sample of the bigger sample
2. This is not a good sample

vocabulary

Documents	a	bigger	good	Is	not	of	sample	the	this
Sentence-1	1	1	0	1	0	1	3	1	1
Sentence-2	0	0	1	1	1	0	1	0	1

Words & representations – Bag of words (Impl.)

- Extract vocabularies
- Compute the occurrences of every word in vocabulary in each sentence
- Generate Term-document matrix

[impl.]: *from Sklearn.feature_extraction.text import CountVectorizer*

Tokenization

- Word tokenization

foo = “Oh God!\n I haven't saved any of it's responses!”

[Oh, God, !, I, have, n't, saved, any, of, it, 's, responses, !]

- Sentence tokenization

bar = “ Sent tokenize knows that time period from 10 a.m. to 1 p.m. are not sentence boundaries. neither are the names G.H.Hardy and J.J.Thompson. you can even start the sentence without Caps”

[“Sent tokenize knows that time period from 10 a.m. to 1 p.m. are not sentence boundaries”,

“neither are the names G.H.Hardy and J.J.Thompson”

“you can even start the sentence without Caps”]

[impl.]: *from nltk import word_tokenize*

Stemming

- Stemming tries to extract the stem word.
- Defined by a set of algorithms like Porter stemmer, Snowball stemmer
- Stem words do not necessarily makes sense

foo = „cylists in all of cities use cycles to cycle the city“

Stems = [cylist, in, all, of, citi, use, cycl, to, cycl, the, citi]

[impl.]: *from nltk import PorterStemmer, SnowballStemmer*

Lemmatization

- Stemming tries to extract the root word.
- Defined by vocabulary of the language
- Lemmas have meanings incontrast to Stem words
- Lemmatization is slower than stemming
- Based on part-of-speech

foo = „it has been used in multiple places“

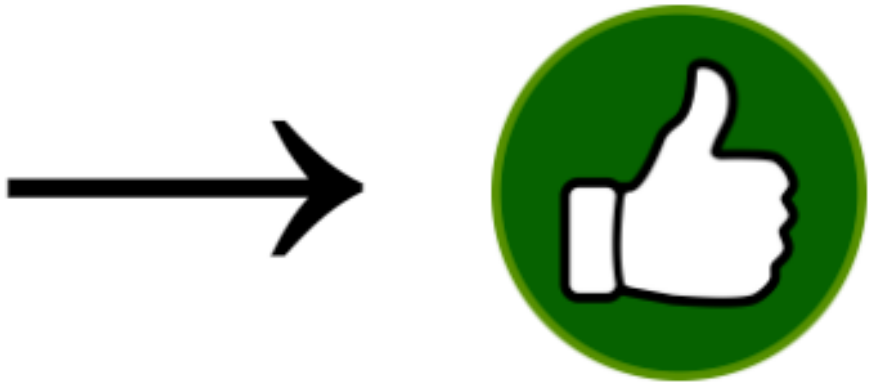
Lemmas= [it, have, be, use, in, multiple, place]

[impl.]: *from nltk.stem import WordNetLemmatizer*

Part - II

Sentiment Analysis

"I love this movie.
I've seen it many times
and it's still awesome."



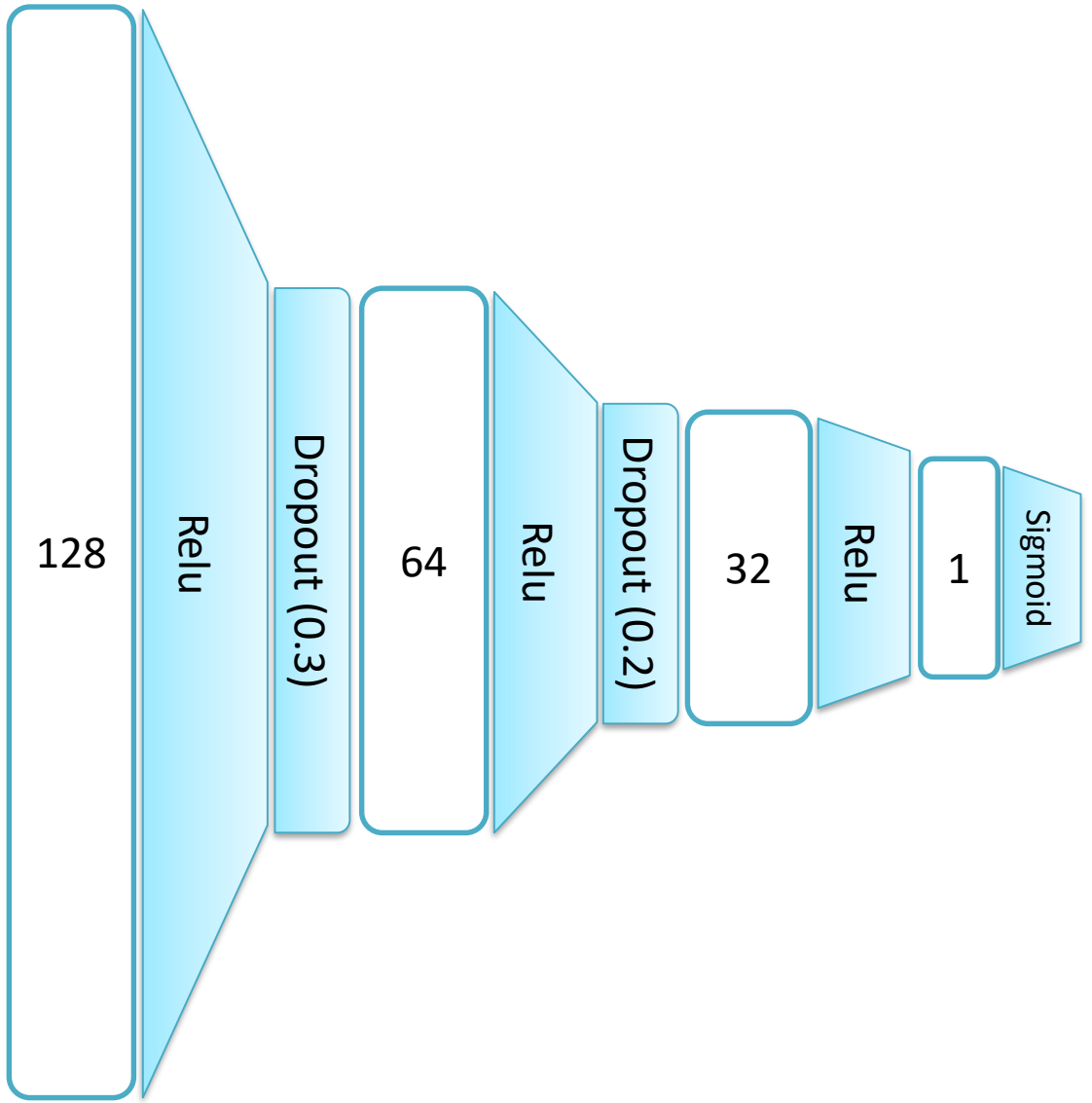
"This movie is bad.
I don't like it it all.
It's terrible."

Sentiment classification

Steps to be followed:

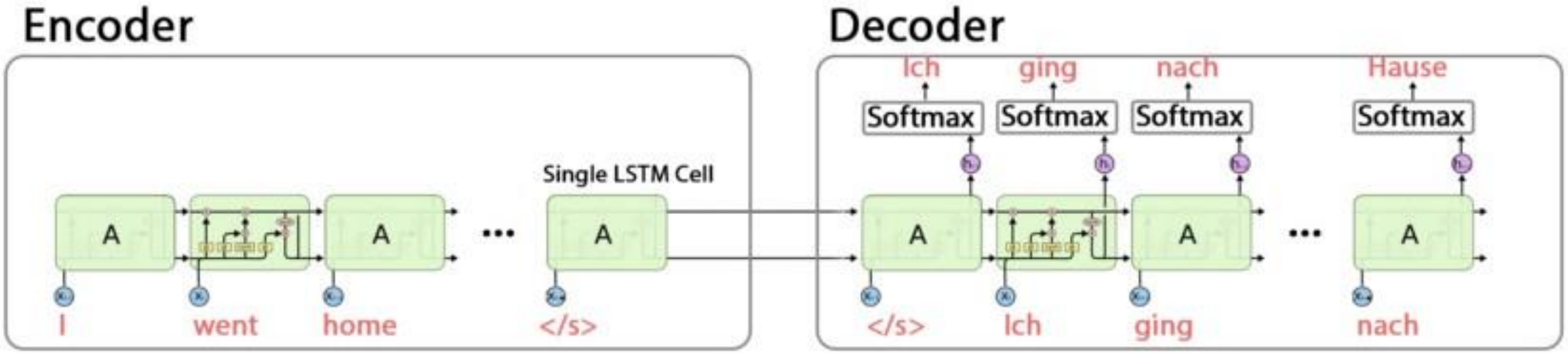
1. Load the dataset
2. Encode the reviews and sentiments
3. Compute Term-document frequency matrix
4. Model training
5. Model prediction

Neural network architecture



Part - III

Machine translation


Encoder
Decoder
Single LSTM Cell
A
A
A
A
A
A
A
Softmax
Softmax
Softmax
Softmax
Ich
ging
nach
Hause
I
went
home
</s>
</s>
Ich
ging
nach

Machine translation

Steps to be followed:

1. Load the dataset (Prepare input and target texts)
2. Encode the characters/tokens as one-hot representation
3. Design the encoder-decoder network
4. Train both the encoder as well as decoder network simultaneously
5. Infer the model using encoder-states and decoder network

Encoder – Decoder structure

Encoder input data:	I	went	home	-	-	-	-
Decoder input data:	\t	Ich	ging	nach	Hause	\n	-
Decoder target data:	Ich	ging	nach	Hause	\n	-	-

Encoder input data shape: (#Sentences, Max length of input sequence, # English vocabulary)

Decoder input data shape: (#Sentences, Max length of target sequence, #German vocabulary)

Decoder target data shape: (#Sentences, Max length of target sequence, #German vocabulary)

Thank you!