Experiment 7

NAME	**Shreya Shetty**
UID	**2019140059**
CLASS	**TE IT**
BATCH	**B**
SUBJECT	**NLP Lab**

AIM: Perform chunking by analyzing the importance of selecting proper features for training a model and size of training.

THEORY:

Feature Engineering:

Feature engineering is one of the most important steps in machine learning. It is the process of using domain knowledge of the data to create features that make machine learning algorithms work. Think machine learning algorithm as a learning child the more accurate information you provide the more they will be able to interpret the information well. Focusing first on our data will give us better results than focusing only on models. Feature engineering helps us to create better data which helps the model understand it well and provide reasonable results.

If we can use these contexts as features and feed them to our model then the model will be able to understand the sentence better. Some of the common features that we can extract from a sentence are the number of words, number of capital words, number of punctuation, number of unique words, number of stopwords, average sentence length, etc. We can define these features based on our data set we are using

Feature Selection:

Feature selection is the process of selecting a subset of the terms occurring in the training set and using only this subset as features in text classification.

Feature selection serves two main purposes:

1. It makes training and applying a classifier more efficient by decreasing the size of the effective vocabulary. This is of particular importance for classifiers that, unlike NB, are expensive to train.

2. Feature selection often increases classification accuracy by eliminating noise features. A noise feature is one that, when added to the document representation, increases the classification error on new data. Suppose a rare term, say arachnocentric, has no information about a class, say China, but all instances of arachnocentric happen to occur in China documents in our training set. Then the learning method might produce a classifier that misassigns test documents containing arachnocentric to

China. Such an incorrect generalization from an accidental property of the training set is called overfitting.

We can view feature selection as a method for replacing a complex classifier (using all features) with a simpler one (using a subset of the features). It may appear counterintuitive at first that a seemingly weaker classifier is advantageous in statistical text classification, but we generally observe that weaker models are often preferable when limited training data are available.

Chunking:

Chunking is a process of extracting phrases from unstructured text, which means analyzing a sentence to identify the constituents(Noun Groups, Verbs, verb groups, etc.) However, it does not specify their internal structure, nor their role in the main sentence. It works on top of POS tagging. It uses POS-tags as input and provides chunks as output.

In short, Chunking means grouping of words/tokens into chunks.

Chunking can break sentences into phrases that are more useful than individual words and yield meaningful results.

Chunking is very important when you want to extract information from text such as locations, person names. (entity extraction)

A sentence typically follows a hierarchical structure consisting of the following components.

sentence → clauses → phrases → words

Groups of words make up phrases and there are five major categories.

- Noun Phrase (NP)
- Verb phrase (VP)
- Adjective phrase (ADJP)
- Adverb phrase (ADVP)
- Prepositional phrase (PP)

Chunking is an analysis of a sentence which identifies the constituents (noun groups, verbs, verb groups, etc.) which are correlated. These are non-overlapping regions of text. Usually, each chunk contains a head, with the possible addition of some function words and modifiers either before or after depending on languages. These are non-recursive in nature i.e. a chunk cannot contain another chunk of the same category.

Some of the groups possible are:

1. Noun Group

2. Verb Group

For example, the sentence 'He reckons the current account deficit will narrow to only 1.8 billion in September.' can be divided as follows:

[NP He] [VP reckons] [NP the current account deficit] [VP will narrow] [PP to] [NP only 1.8 billion] [PP in] [NP September]

Each chunk has an open boundary and close boundary that delimit the word groups as a minimal non-recursive unit.

Hidden Markov Model:
In the mid 1980s, researchers in Europe began to use Hidden Markov models (HMMs) to disambiguate parts of speech. HMMs involve counting cases, and making a table of the probabilities of certain sequences. For example, once you've seen an article such as 'the', perhaps the next word is a noun 40% of the time, an adjective 40%, and a number 20%. Knowing this, a program can decide that "can" in "the can" is far more likely to be a noun than a verb or a modal. The same method can of course be used to benefit from knowledge about the following words.

More advanced ("higher order") HMMs learn the probabilities not only of pairs, but triples or even larger sequences. So, for example, if you've just seen an article and a verb, the next item may very likely be a preposition, article, or noun, but much less likely another verb.

When several ambiguous words occur together, the possibilities multiply. However, it is easy to enumerate every combination and to assign a relative probability to each one, by multiplying together the probabilities of each choice in turn.

It is worth remembering, as Eugene Charniak points out in Statistical techniques for natural language parsing, that merely assigning the most common tag to each known word and the tag "proper noun" to all unknowns, will approach 90% accuracy because many words are unambiguous.

HMMs underlie the functioning of stochastic taggers and are used in various algorithms.

Conditional Random Field:
Conditional random fields (CRFs) are a class of statistical modeling methods often applied in machine learning, where they are used for structured prediction. Whereas an ordinary classifier predicts a label for a single sample without regard to "neighboring" samples, a CRF can take context into account. Since it can consider context, therefore CRF can be used in Natural Language Processing. Hence, Parts of Speech tagging is also possible. It predicts the POS using the lexicons as the context.

In this experiment both algorithms are used for training and testing data. As the size of the training corpus increases, it is observed that accuracy increases. Further, even features also play an important role for better output. In this experiment, we can see that Parts of Speech as a feature performs better than only lexicon as the feature. Therefore, it is important to select proper features for training a model to have better accuracy.

IDE USED: Jupyter Notebook

COLAB LINK:

https://colab.research.google.com/drive/1chz-9XV3EnUfMwD6Yi7WAptjyUYc31Cv#scrollTo=jnGRsCMjQdQv

REFERENCES:

1. https://youtu.be/b4nbE-pG_TM
2. https://www.youtube.com/watch?v=7QmW68C7Aw0
3. https://towardsdatascience.com/a-practitioners-guide-to-natural-language-processing-part-i-processing-understanding-text-9f4abfd13e72
4. https://www.geeksforgeeks.org/nlp-chunking-and-chinking-with-regex/
5. https://www.geeksforgeeks.org/nlp-training-tagger-based-chunker-set-1/