Chien BMI 500 Week 6 Write Up

Frank Chien

October 2021

1 Vectorization

Please see the ipynb file for results of the vectorization exercise. This corresponds to homework problem number 1

2 Part of speech tagging

The basic tagger, which simply tags all tokens as nouns, is the simplest tagging method that can imagined. This tagging system does not take into account context, neither does it take into account what the token is itself. In natural language, nouns are the most commonly used, so by pure chance a randomly selected token is more likely a noun than other parts of speech.

In itself, it is not a very useful tagger. However, when encountering unrecognized tokens that other tagging systems fail to identify, it's reasonable to simply guess it's a noun. Thus it is a reasonable back-off option.

Simply tagging with a basic noun tagger, we arrive at 11.4% of tokens tagged correctly. Please see the attached ipynb file for further details on code and comments.

The unigram tagger or token lookup method takes a look at what the token is, and matches it to a dictionary of most common parts of speech. As written in the ipynb file, it takes n where n is the number of most common tokens to look-up. The unigram tagger gives an accuracy of 45%, 59%, and 73% using n of 50, 100, and 1000 respectively. When the unigram tagger fails to identify a token, the simple default noun tagger can be used instead. When this backoff strategy is applied, the accuracy improves to 56%, 69%, and 79%, respectively. For larger n, the accuracy increases in the unigram tagger. Regardless of how many words we lookup, using the default tagger backoff increases the accuracy, however the improvement decreases with larger n. This makes sense because there are fewer words we are guessing at as n increases.

The bigram tagger utilizes the n-1 word in the training set. The advantage is the bigram tagger will use sentence context to improve its tagging precision. For example, if it see's "to plant" vs "leafy plant", the tagger may reasonably infer that plant in 'to plant' is a verb while the same token acts as a noun in "leafy plant". However, the tagger will only tag if its seen the exact n and n-1 combination, thus there are a lot more words that will go untagged. The bigram tagger therefore offers us more certainty in the part of speech tags, but at the cost of being generalizable in unfamiliar texts. With the bigram tagger alone, 16% of the words were tagged correctly.

We can utilize the relative advantages of each tagger by first starting with the bigram tagger, then backing off to the unigram tagger, and finally backing off to the default tagger. This combination allows 83% of tokens to be tagged correctly, and offers the best performance of any of the strategies described above.

Can lower-casing text effect POS tagging performance? Yes, because in natural language, casing can give information with regards to parts of speech. This happens with proper nouns. Examples include: Seattle Seahawks, Philadelphia Eagles, or Venmo. In cases such as Venmo or Google, capitalization can change the part of speech. Consider "may I venmo you the money?". Here venmo is used as a verb Alternatively, consider "I've read that Venmo is owned by PayPal". Here, Venmo refers to the app and is a noun.

Thus, lowercasing all tokens in the a text can effect POS tagger performance because you may lose information on whether or not a token was used referring to the action it represents or as a proper noun.

3 Possible Components of NLP in Question Answering Systems

Question answering systems will likely need to utilize a couple of NLP strategies. The first is parsing through the question to recognize what the user is asking. To simply the query, stopwords might be removed. Part of speech tagging might be used to help with word sense disambiguation. An example is if a user queries "How do I treat high blood pressure with medications?", we may identify "treat", "high blood pressure", and "medications" as important tokens. Bigram POS tagging may be used to identify "treat" in the verb sense as it was used in "to treat" rather than in the noun sense such as "have a treat". Once the system parses through the query, it may scan a body of literature looking for relevant documents. Single or multi-document summarization methods (though understandably difficult) may be used to return to answer the query.