Reflection:

This lab gave us hands-on experience with Bag of Words, TF-IDF, and Word Embeddings. I learned that Bag of Words is not as reliable as TF-IDF in most situations, but it does have its pros in others. It was nice to learn about vectors and why we need to use them. The bad part about Bag of Words is that it cannot process context and can assign a lot of zeros to words whenever a lot of fluff is used. When needing the type of representation to be more sophisticated, it is better to go for TF-IDF. Term Frequency-Inverse Document Frequency can filter out the fluff words and focus on the ones that matter. It is useful to get rid of all that extra noise to give better results in classification and keep search results relevant to what is needed. It is similar to word embeddings, but not quite the same. TF-IDF is more sparse while word embeddings are dense! Word Embeddings is better for contextual situations while TF-IDF is based on word frequency. The cool part about word embeddings is that it can extract the word based on when and where it was used before!

To go into more detail about the pros and cons, I will go into each with more background. When it comes to bags of words, the pros are that it is simpler to use and explain to beginners. It has more sparsity meaning the vector space contains mostly zeros. It also can scale a lot of larger data sets. A few cons are that the order of words does not get taken into consideration along with there being a lack of semantic meaning.

Let's talk about Term Frequency-Inverse Document Frequency, "term frequency" being in the name due to its ability to capture how many times a word shows up and IDF meaning increase of weight for rare words found. Both of these work together to improve machine learning with text data along with helping retrieve the most accurate search results pertaining to the topic provided. TF-IDF also uses sentiment analysis which is able to pick up on the tone

being used such as positive, negative, or neutral meanings. The formulas for these work together by multiplying term frequency with the inverse document frequency. A con of using TF-IDF is that it cannot associate a meaning to the word. As well as if a word is extra rare in a document, it can give more importance to it than is actually needed. Unfortunately, it also lacks the ability to associate context with how a word is being used and similar to BOW, it does not capture the order of words being used.

Word embedding is the better option if you are needing the context to be acknowledged and semantic relationships to be understood. With the continuous vector space used, each word is able to be assigned to a vector surrounded by other similar contextual words. It is also the one out of the three we learned to be based on neural networks. This is the preferred method when needing a deeper understanding of data. With that being said, it is costly in the sense of how much memory is needed to retain the vectors.

Overall, all three function pretty great as long as they are used in the proper environments. The pros outweigh the cons for them, but in a given situation you can be faced with mainly cons. It is important to research the end goal of a project so that you know when and where to use each one. This lab was helpful in providing hands-on coding and seeing how it can be helpful to use one over the other.