|  |
| --- |
| Advantages & Disadvantages of Encoding  Assignment |
|  |



**1. One Hot Encoding:**

In this encoding the a word is represented as a vector whose dimension are the length of the dictionary or corpus , in which the value of the position of the current word is 1 , and rest are 0 .

* Create a dictionary based on the corpus (vocabulary) and create a mapping of words and indexes
* The length of the word vector is the length of the dictionary .
* One-Hot To encode the text, the resulting matrix is sparse matrix(Sparse matrix) .
* Ideally for categorical features with less than 10 categories (max 50 )

**Advantages:**

1. Easy to implement.
2. It preserves all information about the categories and doesn’t introduce any ordinal relationship.

**Disadvantages:**

1. The vector representations of different words are orthogonal to each other and cannot measure the relationship between different words.
2. The code can only reflect whether a word appears in a sentence and cannot measure the importance of different words.
3. One-Hot Encoding the text yields a high-dimensional sparse matrix that wastes computational and storage resources.
4. Context is not set .
5. Leads to high dimensionality problem .

**2. Bag Of Words(BOW):**

Each word is independent of each other. The words are put into a "bag" and the frequency of occurrence of each word is counted.

* The length of the encoded vector is the length of the dictionary;
* The encoding ignores the order in which the words appear;
* In the vector, the value of the index position of the word is the number of times the word appears in the text; if the word at the index position does not appear in the text, the value is 0.

**Advantages:**

1. Simple and intuitive to convert from word to numbers .

**Disadvantages:**

1. The code ignores the location information of the word. The location of the word is not the same as the semantics (such as "cat loves to eat mouse" and "mouse loves cat" code) .
2. It cannot distinguish common words (such as "I", "Yes", "", etc.) and keywords (such as: " Natural language processing", "NLP The importance of "etc." ) in the text .
3. It considers the number of times each word appears in a sentence but does not consider their order, which means we lose the context of the words.

**3. N-Grams :**

A n-gram is basically a sequence of arbitrary words, having a length of n. For instance, “Thank You” is a 2-gram (a bigram), “useful in NLP” is a 3-gram (a trigram), “NLP is quite cool” is a 4-gram.

* The N-gram model is a statistical language model that estimates the probability of the next word in a sequence based on the previous N-1 words.
* It works on the assumption that the probability of a word depends only on the preceding N-1 words

**Advantages:**

1. Simplicity: N-grams are intuitive and relatively simple to understand and implement.
2. Low Memory Usage: Require minimal memory for storage compared to more complex models.
3. It is computationally efficient and can be used for real-time applications.
4. It can handle large amounts of text data and provide accurate results.

**Disadvantages:**

1. Limited Context: N-grams have a finite context window.
2. Sparsity: As N increases, the number of possible N-grams grows exponentially, leading to sparse data and increased computational demands.
3. N-grams cannot deal Out Of Vocabulary (OOV) words. It works well with the words present in the training set. In the case of an Out Of Vocabulary (OOV) word, n-grams fail to tackle it
4. A count may not necessarily indicate importance to text or entity.
5. N-gram language model does not capture the semantic meaning of words and cannot handle the ambiguity of language.

**4. TF-IDF ( Term Frequency-Inverse Document Frequency )**

Each word is independent of each other. The words are put into a "bag" and the frequency of occurrence of each word is counted.

* TF (Term Frequency ）: The frequency with which a word appears in the current text .
* IDF (Inverse Document frequency ）: Reverse text frequency. Text frequency refers to the proportion of text containing a word in the entire corpus. The inverse text frequency is the reciprocal of the text frequency.

**Advantages:**

1. Simple implementation, easy to understand algorithm and strong explanatory.
2. You can easily compute the similarity between 2 documents using it.
3. For more frequent words, IDF will be small; and keywords (such as: "natural language deal with","NLP "etc." ) will appear less and have large IDF The value. So TF-IDF You can filter out some common, irrelevant words while retaining the important words of the article.

**Disadvantages:**

1. The position information of the word cannot be reflected. So ignores word order .
2. TF-IDF Severely dependent on the corpus. So it is necessary to select a high-quality corpus for training.
3. Overweighting of rare words: Words that occur rarely in a large corpus may have a higher TF-IDF score, but they may not necessarily be relevant to the document or query.
4. Not considering context
5. Ignoring semantic relationships: TF-IDF does not account for semantic relationships between words, such as synonymy, antonymy, and hyponymy, which can lead to poor retrieval results.