

Artificial Intelligence

Pattern classification

1. Implement a function that will extract word features for a string x (`extractWordFeatures(x)`). Words should be delimited by whitespace characters only. You should return a feature vector as Python dictionary.

Example: "I am what I am" --> {'I': 2, 'am': 2, 'what': 1}

2. From Wikipedia, an n-gram is a contiguous sequence of n items from a given sample of text or speech. For this problem you will create a function that takes a string x and returns a feature vector consisting of all n-grams of x without spaces mapped to their n-gram counts.

Example: ($n = 3$) "I like artificial intelligence" --> {'Ili': 1, 'lik': 1, 'ike': 1, ...}

3. For this problem you will have to write your own implementation of a simple 2-D k-means algorithm (`kmeans(examples, K, maxIters)`). You can read more about the algorithm [here](#). You should initialize your k cluster centers to random elements of examples. Examples is a list of examples where each example is a 2-D vector, K is the number of desired clusters and should be $0 < K \leq |examples|$, and `maxIters` is a maximum number of iterations to run your algorithm, but you should terminate early if the converges is reached. Remember that you are not allowed to use libraries like Scikit-learn to do this problem.