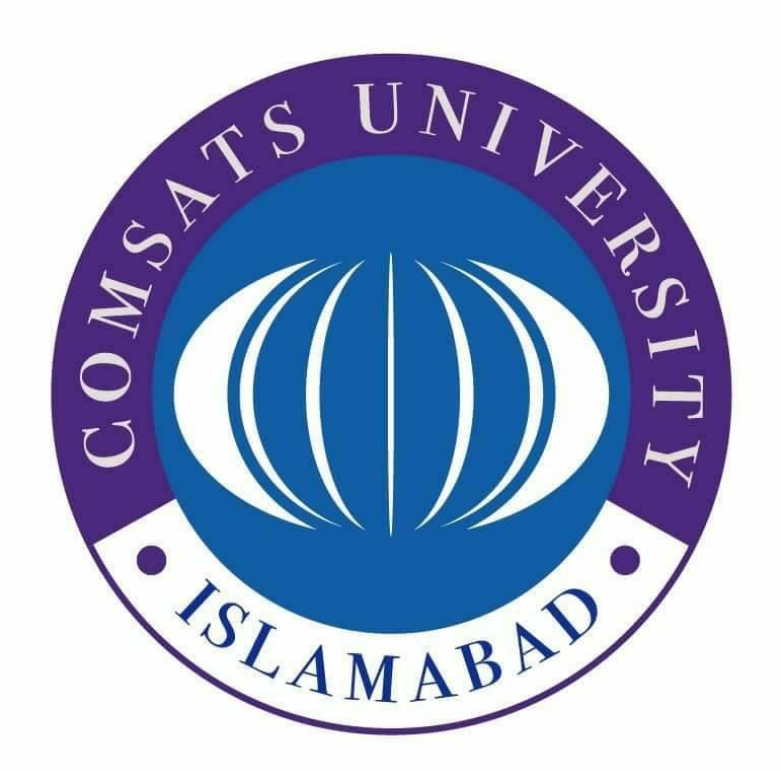

Assignment-05

INTRODUCTION TO DATA SCIENCE (IDS)

SUBMITTED BY:

Ahmad Nawaz Khan

CIIT/SP20-BCS-008/LHR

SUBMITTED TO:

Sir Muhammad Sharjeel

ANSWER No.1:

Bag of Words (BoW) model:

The BoW model represents a document as a bag of words, ignoring the word order and only considering the presence or absence of each word.

For the given sentences, the BoW model would be:

S1: {sunshine, state, enjoy}

S2: {brown, fox, jump, high, run}

S3: {sunshine, state, fox, run, fast}

Term Frequency (TF) model:

The TF model represents a document as a vector of the frequencies of each term in the document.

For the given sentences, the TF model would be:

S1: {sunshine: 2, state: 1, enjoy: 1}

S2: {brown: 2, fox: 2, jump: 1, high: 1, run: 1}

S3: {sunshine: 1, state: 1, fox: 1, run: 1, fast: 1}

Inverse Document Frequency (IDF) Model:

The IDF model represents the importance of each term in a collection of documents. It is calculated as the logarithm of the total number of documents divided by the number of documents containing the term.

For the given sentences, the IDF model would be:

IDF(sunshine) = log(3/2) = 0.405

IDF(state) = log(3/2) = 0.405

IDF(enjoy) = log(3/1) = 0.693

IDF(brown) = log(3/2) = 0.405

IDF(fox) = log(3/2) = 0.405

IDF(jump) = log(3/1) = 0.693

IDF(high) = log(3/1) = 0.693

IDF(run) = log(3/2) = 0.405

IDF(fast) = log(3/1) = 0.693

TF-IDF Values:

The TF-IDF value of a term in a document is the product of its TF and IDF values.

For the given sentences, the TF-IDF values would be:

S1:

TF-IDF(sunshine) = 2 * 0.405 = 0.810

TF-IDF(state) = 1 * 0.405 = 0.405

TF-IDF(enjoy) = 1 * 0.693 = 0.693

S2:

TF-IDF(brown) = 2 * 0.405 = 0.810

TF-IDF(fox) = 2 * 0.405 = 0.810

TF-IDF(jump) = 1 * 0.693 = 0.693

TF-IDF(high) = 1 * 0.693 = 0.693

TF-IDF(run) = 1 * 0.405 = 0.405

S3:

TF-IDF(sunshine) = 1 * 0.405 = 0.405

TF-IDF(state) = 1 * 0.405 = 0.405

TF-IDF(fox) = 1 * 0.405 = 0.405

TF-IDF(run) = 1 * 0.405 = 0.405

TF-IDF(fast) = 1 * 0.693 = 0.693

ANSWER No.2:

For S1 and S3, the vectors can be represented as follows:

S1: [2, 1, 1]

S3: [1, 1, 1, 1]

We can calculate the cosine similarity by taking the dot product of the vectors and dividing it by the product of the vectors' magnitudes.

The dot product of the vectors is:

(2 * 1) + (1 * 1) + (1 * 1) = 4

The magnitudes of the vectors are:

|S1| = sqrt((2 * 2) + (1 * 1) + (1 * 1)) = 2.4495

|S3| = sqrt((1 * 1) + (1 * 1) + (1 * 1) + (1 * 1)) = 2

So, the cosine similarity between S1 and S3 is:

Cosine Similarity = (4) / (2.4495 * 2) = **0.8165**

This means that S1 and S3 are very similar, with a cosine similarity of 0.8165.