

Homework – 3

Content covered: Spark Streaming

Due: 5th May, 2024

Homework Overview:

This assignment will provide you with hands-on experience in writing and executing Spark streaming code. You will begin by installing and configuring Spark, then proceed to write a word co-occurrence program. Following the implementation phase, you will analyze the data flow within Spark applications, gaining insights into how Spark processes and manipulates data across distributed computing environments. Through this comprehensive approach, you'll not only develop practical programming skills but also deepen your understanding of distributed data processing concepts using Spark streaming.

General Homework Requirements:

- **Work Environment:** This homework can be written in Pyspark /Python or Scala.
- **Programming:** You can use Jupyter Notebook, Jupyter Lab or Google collab or any Scala and python IDE
- **Academic Integrity:** You will get an automatic F for the course if you violate the academic integrity policy.
- **Teams:** This homework is an individual assignment. You are not permitted to work with anyone else on this assignment. All work submitted must be yours and yours alone.

Submission Format:

1. **Source Directory:** All input data files, and code implementations should be organized within a specific directory named "src." This directory will contain both the input datasets and the code files required for the assignment.
2. **Report:** Prepare a comprehensive report containing answers to all questions posed in the assignment. Each answer should include suitable proofs or evidence to validate the authenticity of your submission and demonstrate that the outputs are legitimate. This report should be well-structured and provide clear explanations for each question, along with any necessary supporting materials ,documentation or screenshots of Code

By adhering to these refined requirements, you will ensure that your submission is well-organized, thoroughly documented, and adequately substantiated, thereby demonstrating your proficiency in completing the assignment successfully. **Failure to adhere to the specified submission format will result in a deduction of 3 marks. All submissions must follow the prescribed structure to ensure consistency and clarity. Submissions must be made on Brightspace by 5th May @ 11:59 PM. No late submissions will be accepted. Even a delay of 1 minute will result in a score of 0 for the assignment. It's recommended to upload your HW-3 at least a few hours before the due date and time to avoid any last-minute issues.**

Setup:

- To prepare your development environment for this homework you must first install and set up PySpark. To install PySpark, follow the instructions here: https://spark.apache.org/docs/latest/api/python/getting_started/install.html
- Install SparkConf, SparkContext and StreamingContext
- Run the python file named “paragraph_generator.py” and it will generate a text file named “paragraph.txt”.
- Install netcat
<https://nmap.org/download.html>
- You must write a word co-occurrence program that will find the co-occurrences of words in the text of ‘paragraph.txt’.
- For continuous streaming input , you will use the ‘paragraph.txt’ as input stream through the TCP socket port 9999

Questions:

1. Implement word co-occurrence program.

[6X15=90 Marks]

- **Step 1:** Create a local StreamingContext with two execution threads and batch interval of 1 second.
- **Step 2:** Create a DStream that represents streaming data from a TCP source (localhost:9999). [hint: use Netcat]
- **Step 3:** Split each line into words, normalize it to lowercase, and remove punctuation.
- **Step 4:** Create bigrams from a list of words; If you want to know how to create bigrams please refer [here](#)
- **Step 5:** Apply sliding window transformation to generate bigrams within a window. (hint: window length=3 and sliding interval=2)
- **Step 6:** Count the occurrences of each bigram and print word co-occurrence counts.

2. Comprehensive Report.

[10 Marks]

- Write a comprehensive 500-word review of the [paper](#) on DStream named “**Discretized Streams: Fault-Tolerant Streaming Computation at Scale**”. This paper is available on Brightspace under the resources tab in Spark_resource section.
- Usage of any other online sources other than provided in this course will result in an automatic 0 in this assignment.
- Usage of any Artificial Intelligence tools to generate text will result in a -5 in the course