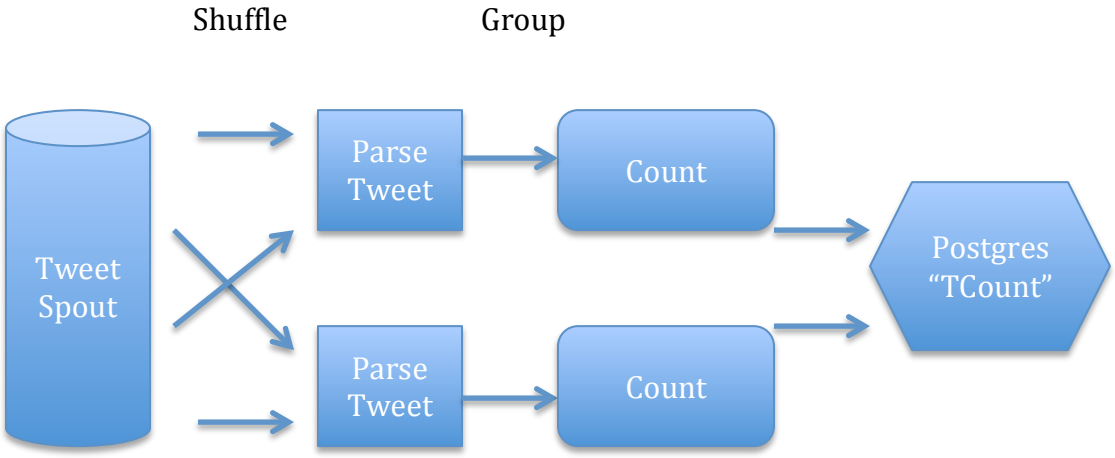
Introduction

The tweetcount application counts the number of words in a live twitter stream. The application diagram is as follows:

Architecture



Tweetcount Topology

The table in the TCount database is called "`tweetwordcount`".

Description

The tweets.py contains the spout for the program. It connects to twitter and consumes the live tweets. The spout uses tweepy to communicate with twitter. These tweets are then sent to the parse bolt. A shuffle grouping is used between the spout and the parse bolt.

Once the tweets have been parsed and broken into words in the parse bolt, the words are sent to the wordcount bolt. The transfer here is grouped by the words. This is to ensure an accurate count of the words. Once the words have been counted they are written to the tweetwordcount table in the TCount database.

In the wordcount bolt, we check if the database exists. If it doesn't, we create it. In this database we check if the tweetwordcount table exists. If it doesn't we create it too. Once everything is set up we write all the counts to the database. The table contains a column for Words and one for Count. The wordcount bolt uses psycopg2 to communicate with postgres.

The project can be run by the command "sparse run –n tweetwordcount"

Directory Structure

The project is called “tweetcount” and the folder structure for the project looks like

Directory	Descriptions
tweetcount/src/spouts	Contains the Spout - **`tweets.py`**
tweetcount/src/bolts	Contains the Bolts - **`parse.py`** & **`wordcount.py`**
tweetcount/topologies	Contains the topology file - **tweetwordcount.clj**