

Business Intelligence Technologies: Practical 5: Text Mining

Weighting: 4% Due date: 23:59pm Friday 6th January 2023

A hotel has collected feedback from its customers and would like you to do some text mining to see what insights can be extracted from this data, eg are customers generally happy or not? If not, what kind of things are customers unhappy about? A manager has gone in and annotated each feedback with a 'positive' or 'negative' classification and has asked whether it would be possible to put together a model that could classify feedback in the future, based on this test data. The manager has also heard that 'sentiment analysis' can be undertaken on this data and wonders whether this might be useful.

Using the hotelfeedback dataset, carry out **two** data mining processes on this data. Annotate **each operator** to explain what it does, what parameters you are using and why, and then provide an analysis of the output.

(Processes may be classification, association rule mining, sentiment analysis, classification, clustering etc.)

Assessment Criteria:

Correct and documented processes following the CRISP methodology, showing sound understanding of the text mining methods used: 2 marks
Documentation and evaluation of results: 2 marks

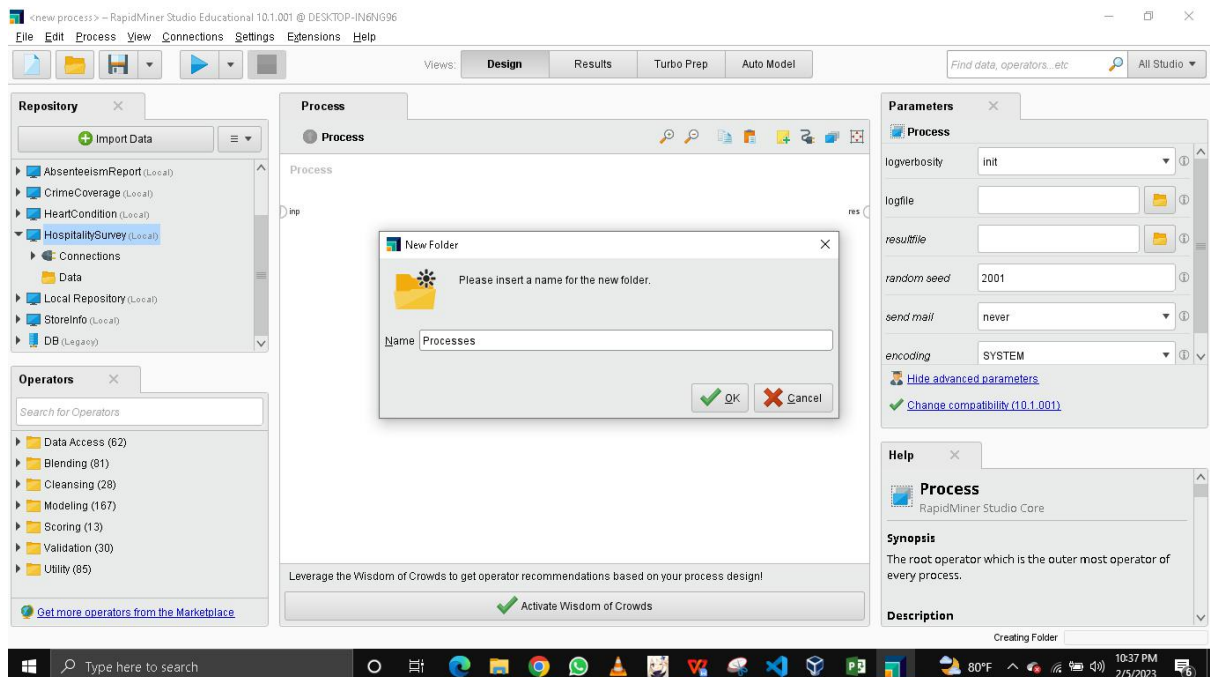
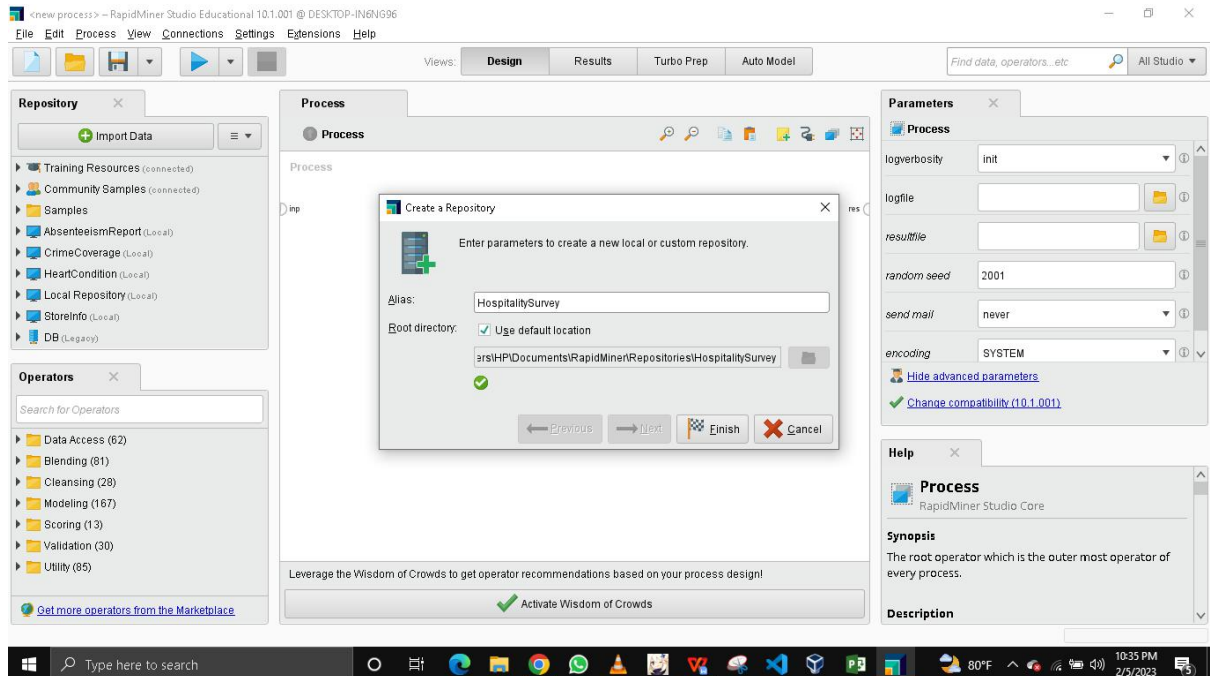
Attribute information:

The data is contained in the 'hotelfeedback' excel file which contains two columns,

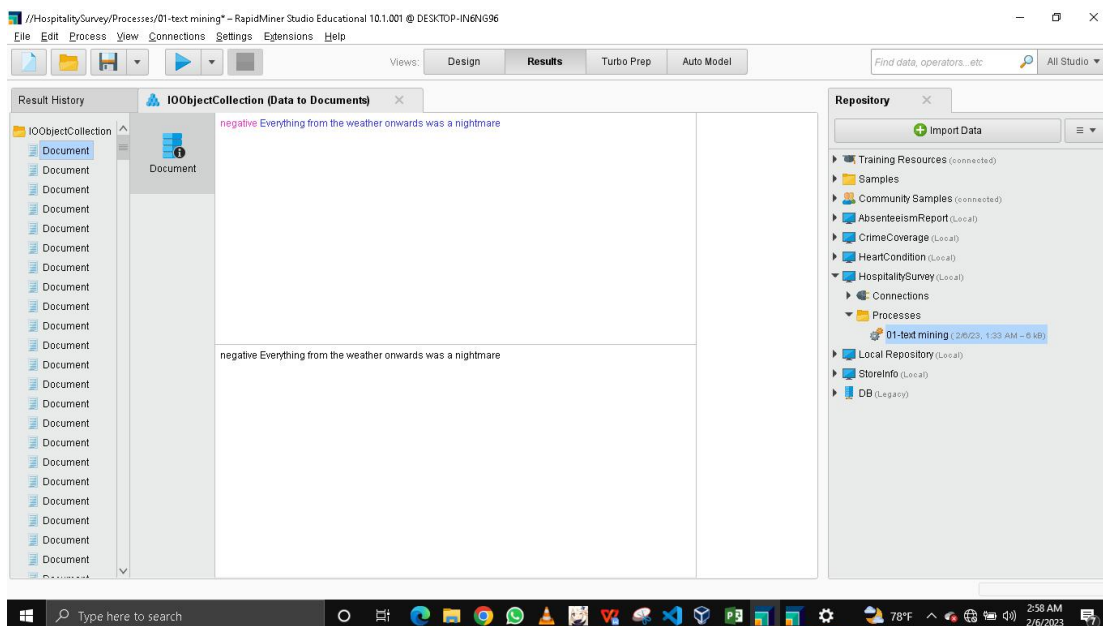
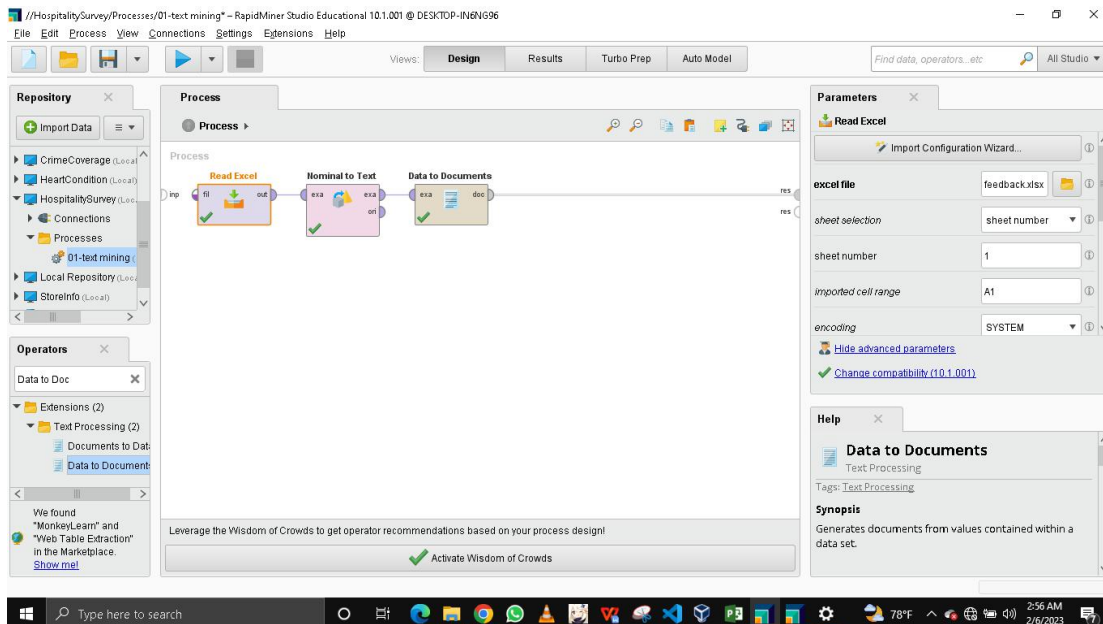
Column1: contains the classification of the feedback (either positive or negative)
Column2: contains the feedback text

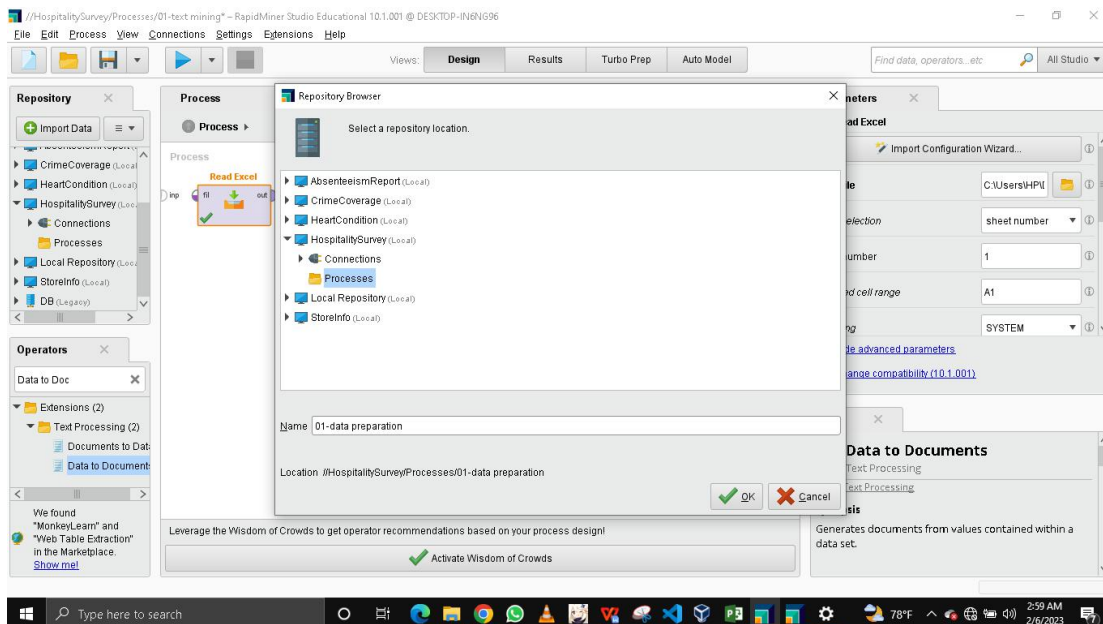
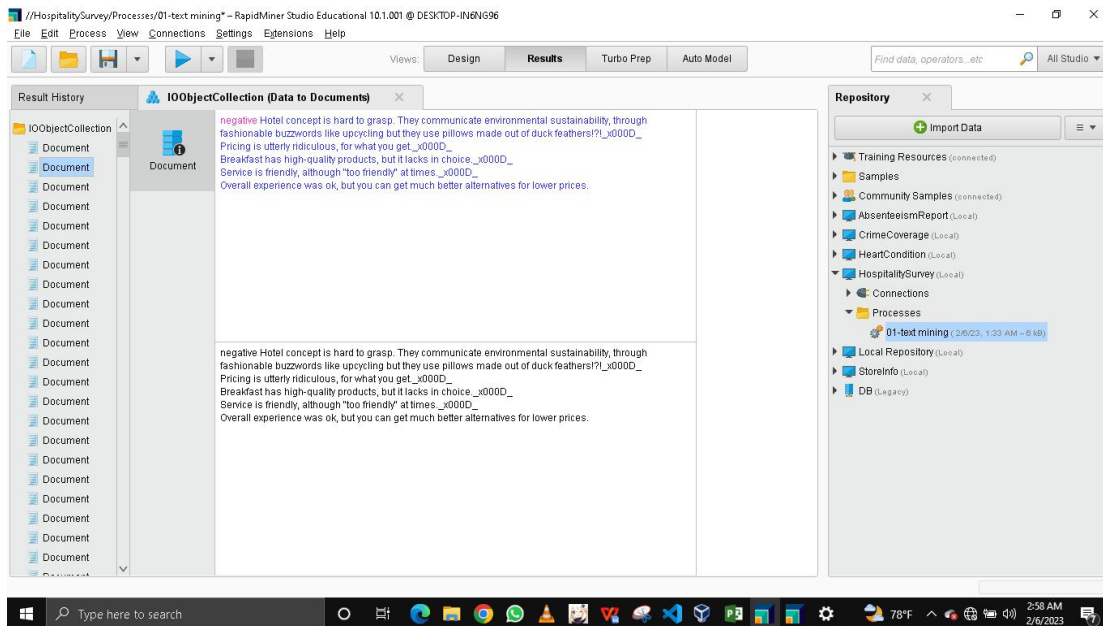
Practical Documentation

1. Created a Repository labeled "HospitalitySurvey" and created a subfolder named "Processes"

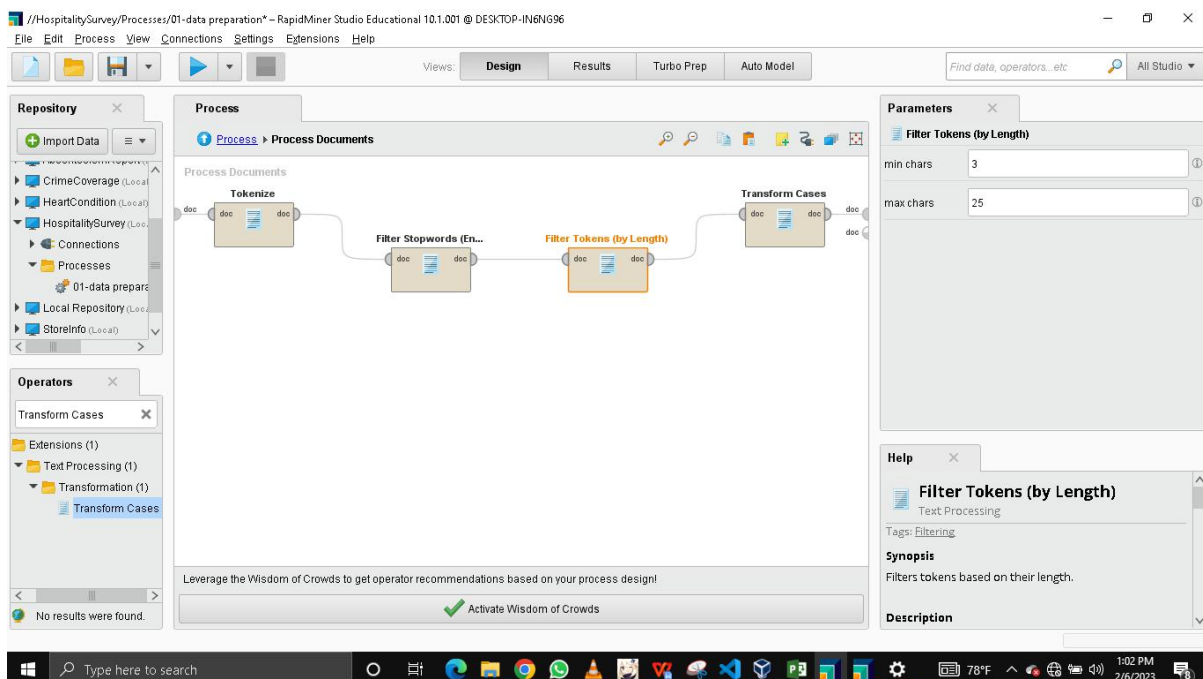
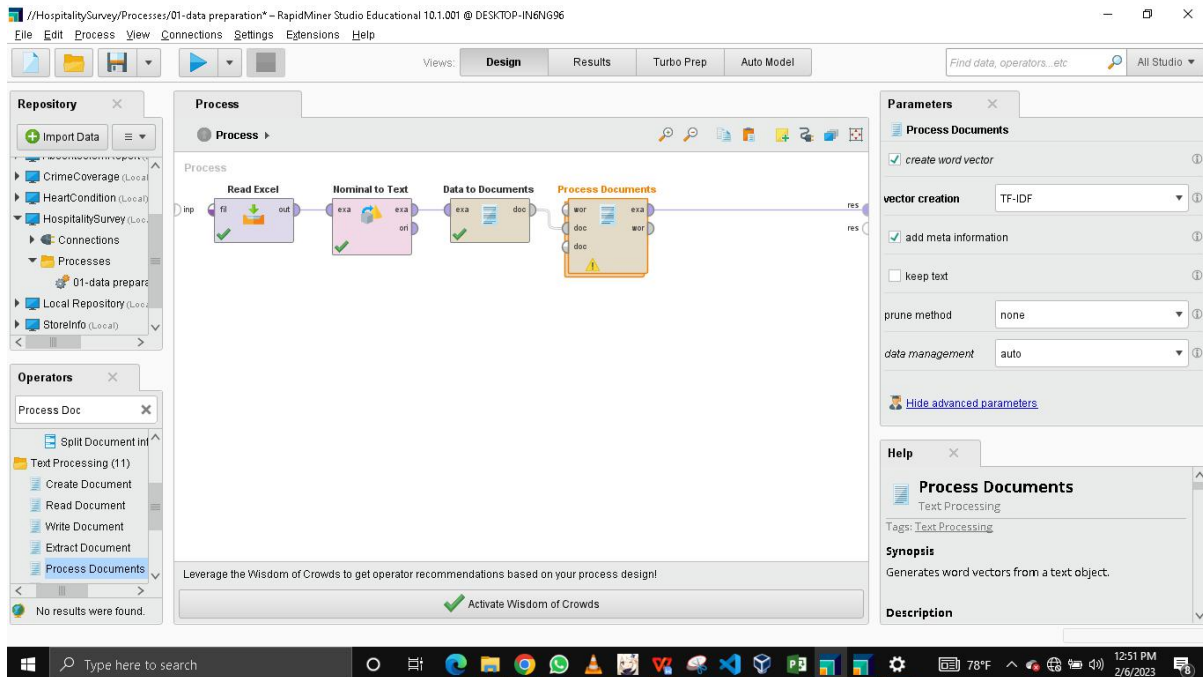


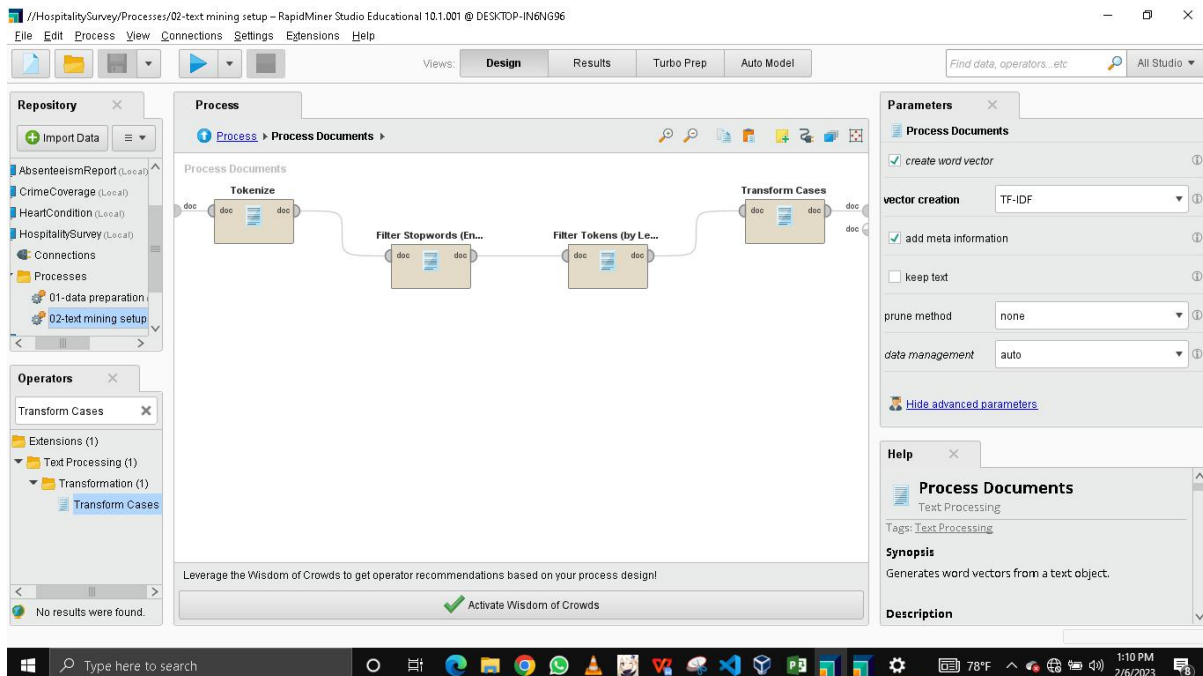
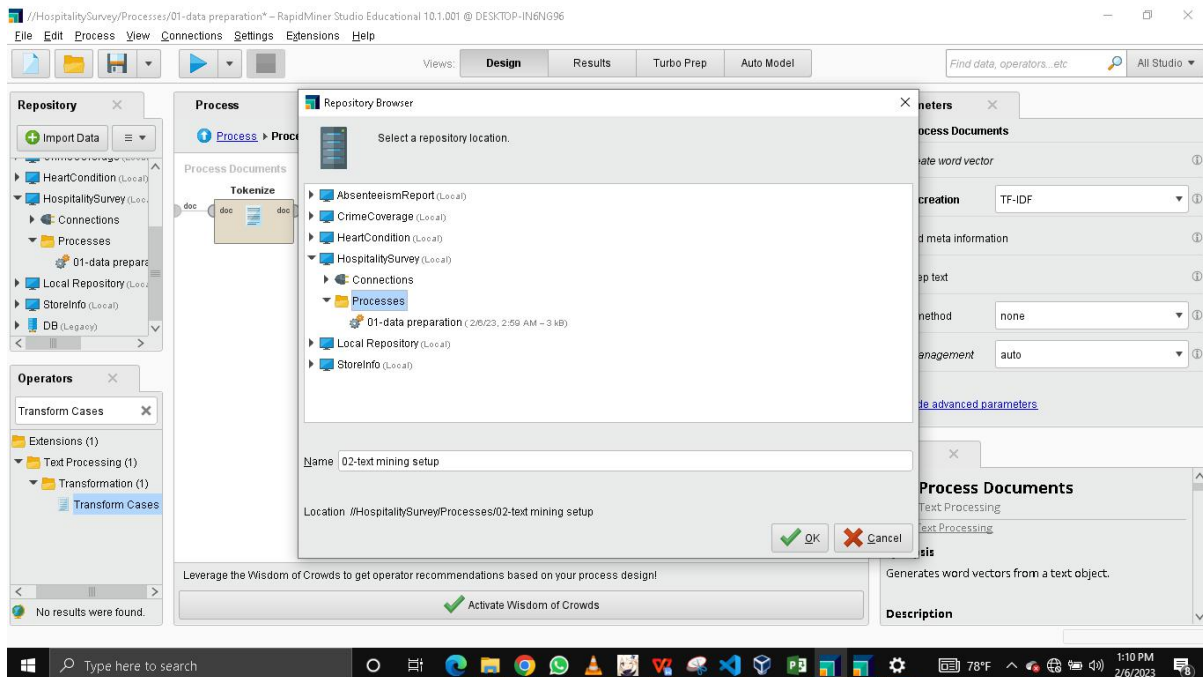
2. Used the “Read Excel” operator to read the hotelfeedback table and connected it to the “Nominal to Text” and “Data to Documents” operators respectively and saved the process as “01-data preparation”, given the customer feedback is stored in an excel sheet and needs to be prepared in the form of text documents prior to being processed for text mining



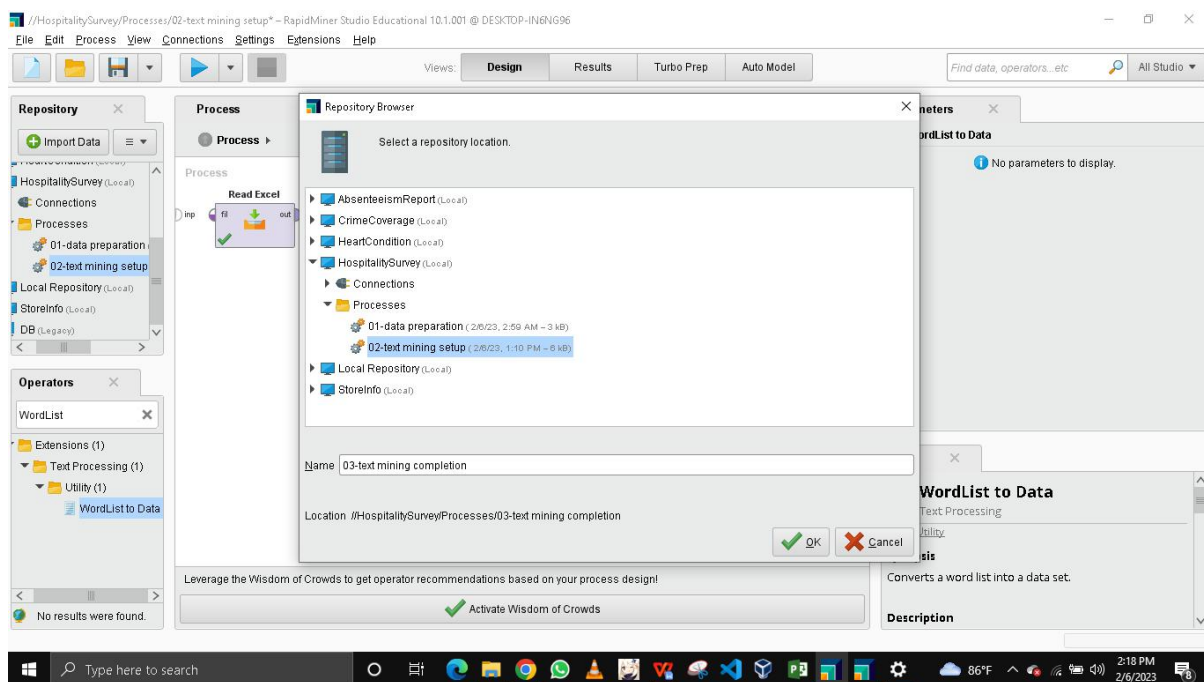
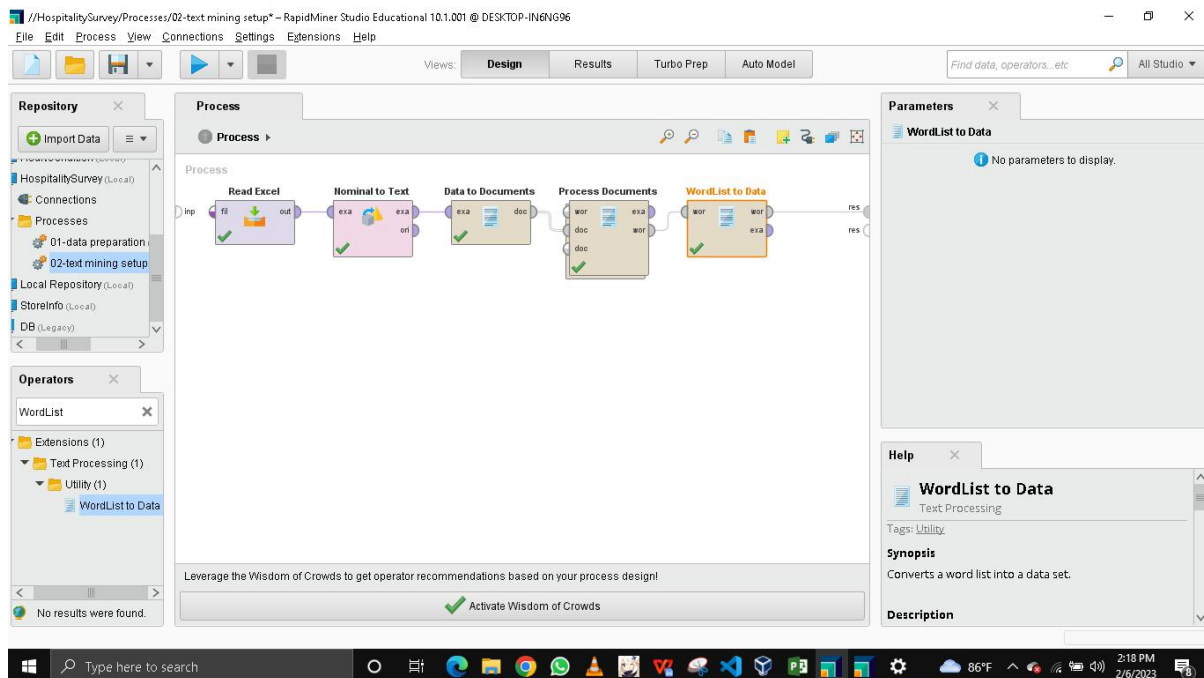


3. Fed the prepared data into the “Process Documents” operator which opens up into a sub-process window that should consist of the following text mining operators such as “Tokenize”, “Filter Stopwords (English)”, “Filter Tokens (by Length)” and “Transform Cases” respectively and to set up the text mining process and minimize the recurrences of the same words in different cases; saving the process as “02-text mining setup”





4. Exited the sub-process window and passed the mined text through the “WordList to Data” operator to get a tabulated version of the text mining results.



The screenshot displays the RapidMiner Studio interface with a process design canvas. The process flow is as follows: **Read Excel** (inputting 'feedback.xlsx', sheet 'sheet number', range 'A1') → **Nominal to Text** → **Data to Documents** → **Process Documents** → **WordList to Data**. The left sidebar shows the 'Repository' with 'HospitalitySurvey (Local)' and 'Processes' containing '01-data preparation', '02-text mining setup', and '03-text mining completion'. The 'Operators' panel shows 'WordList' under 'Extensions (1)' and 'WordList to Data' under 'Utility (1)'. The right sidebar shows the 'Parameters' for 'Read Excel' and a 'Help' section for the 'Read Excel' operator.

Repository: **HospitalitySurvey (Local)**, **Connections**, **Processes** (01-data preparation, 02-text mining setup, 03-text mining completion), **Local Repository (Local)**, **StoreInfo (Local)**.

Process: **Read Excel** (excel file: feedback.xlsx, sheet selection: sheet number, sheet number: 1, imported cell range: A1, encoding: SYSTEM) → **Nominal to Text** → **Data to Documents** → **Process Documents** → **WordList to Data**.

Parameters: **Read Excel** (Import Configuration Wizard..., excel file: feedback.xlsx, sheet selection: sheet number, sheet number: 1, imported cell range: A1, encoding: SYSTEM, Hide advanced parameters, Change compatibility (10.1.001)).

Help: **Read Excel** (RapidMiner Studio Core, Tags: Load, Import, Read, Data, Files, xls, xlsx, Microsoft, Spreadsheets, Datasets, Synopsis: This operator reads an ExampleSet from the specified Excel file).

Text Mining Results

The screenshot displays the 'Results' view of the 'WordList (Process Documents)' operator. The results are presented in a table with columns: Word, Attribute Name, Total O..., and Docum....

Word	Attribute Name	Total O...	Docum...
able	able	3	3
abou	abou	1	1
absolute	absolute	1	1
absolutely	absolutely	13	13
absorb	absorb	1	1
abysmal	abysmal	1	1
accept	accept	1	1
acceptable	acceptable	1	1
accepted	accepted	1	1
accident	accident	1	1
accommodating	accommodating	1	1
accommodation	accommodation	1	1
accosted	accosted	1	1
account	account	2	2
achieved	achieved	1	1

The right sidebar shows the 'Repository' with 'Import Data' and a list of data sources: **Samples** (AbsenteeismReport (Local), CrimeCoverage (Local), HeartCondition (Local), HospitalitySurvey (Local), Local Repository (Local), StoreInfo (Local), DB (Legacy)).

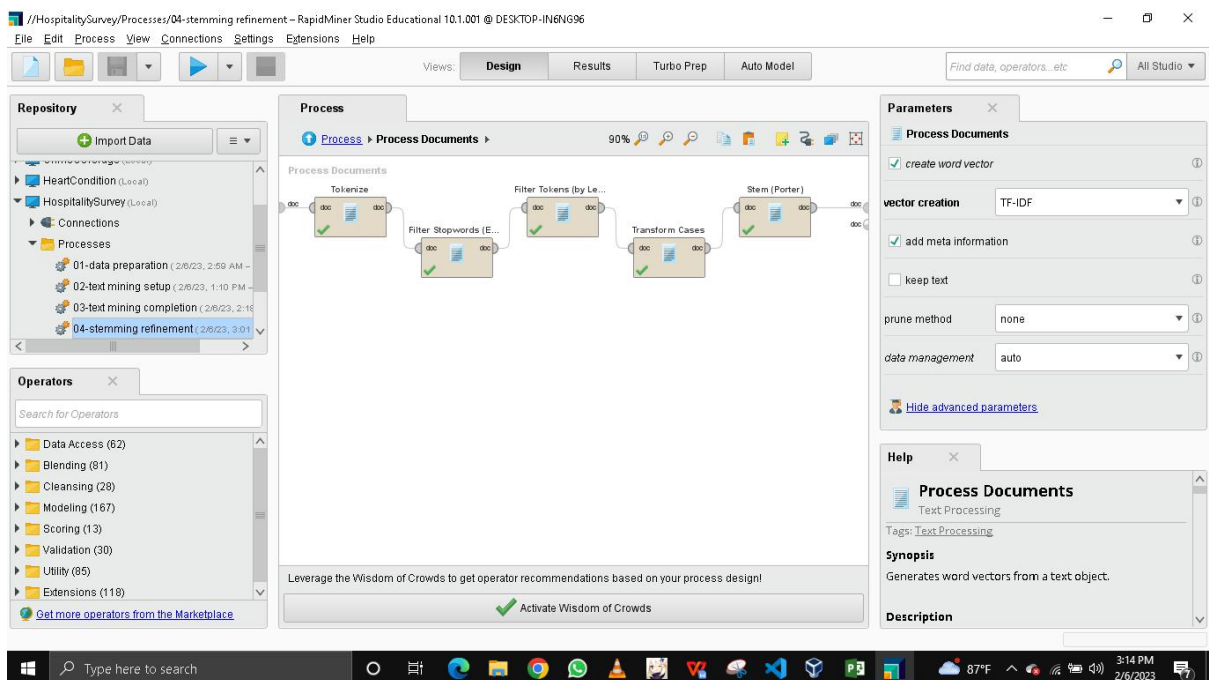
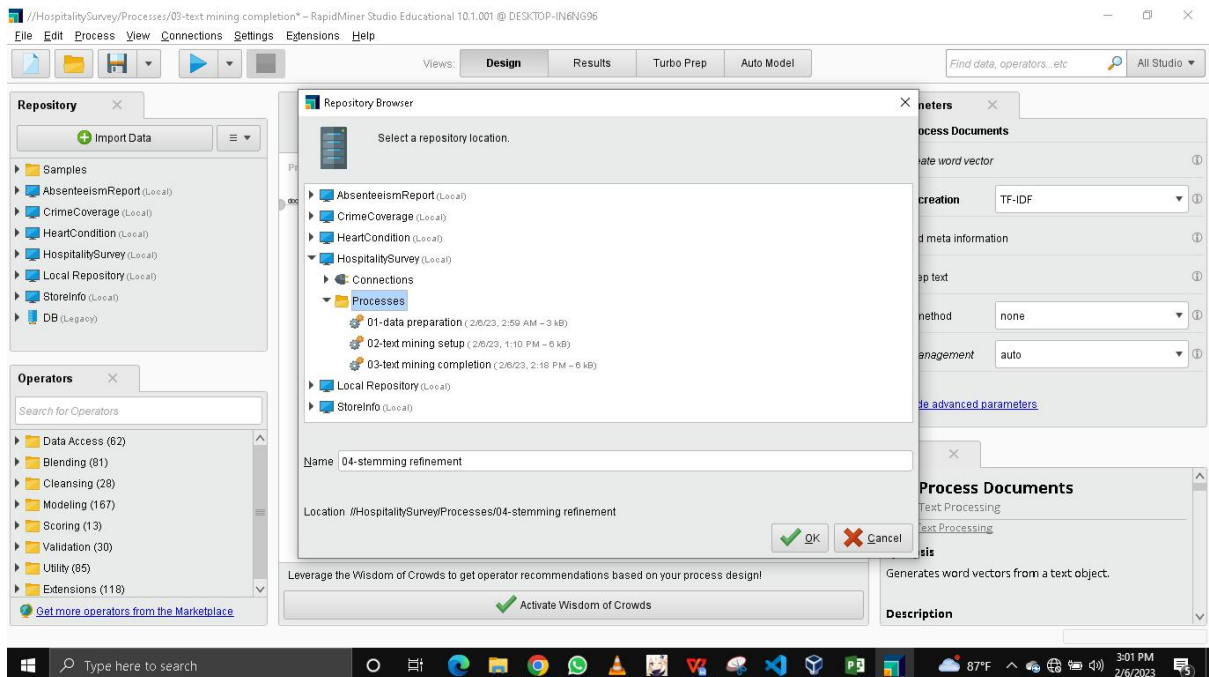
5. Refined the process through the addition of the “Stem (Porter)” operator within the sub-process window of the “Process Documents” operator and saved the process as “04-stemming refinement”

The screenshot displays the RapidMiner Studio interface. The main workspace shows a workflow titled 'Process Documents' with the following steps: 'Tokenize', 'Filter Stopwords (E...)', 'Filter Tokens (by Length)', 'Transform Cases', and 'Stem (Porter)'. The 'Filter Tokens (by Length)' operator is selected, and its parameters are shown on the right: 'min chars' is 3 and 'max chars' is 25. The 'Help' panel for 'Filter Tokens (by Length)' is also visible, showing its synopsis and description. The bottom status bar indicates the system time as 2:55 PM on 2/6/2023.

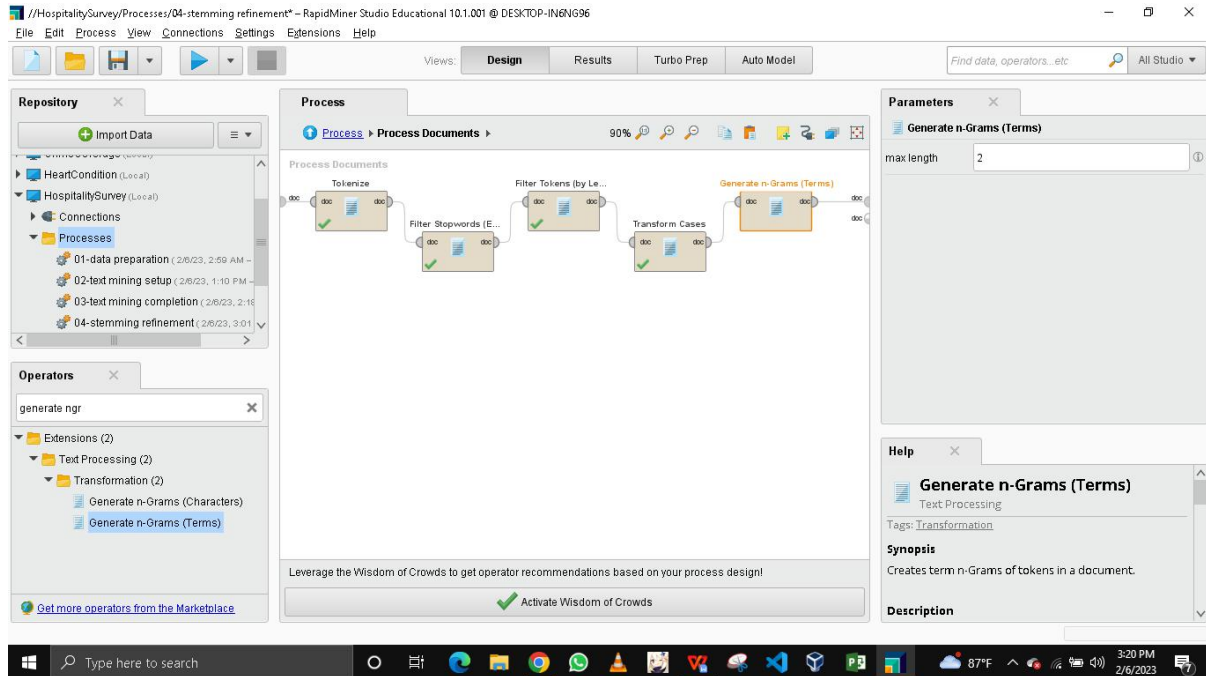
Compared to the text mining results above the results of the refined process using the “Stem (Porter)” operator contains reduced recurrences of words with common root words as a single term such as the total occurrences of the words “absolute(1)” and “absolutely(13)” as a shared total occurrence of the root word “absolute(14)”

The screenshot shows the 'Result History' panel in RapidMiner Studio, displaying the output of the 'WordList (Process Documents)' operator. The results are presented in a table with the following columns: 'Word', 'Attribute Name', 'Total O...', and 'Docum...'. The table lists various words and their occurrences across documents.

Word	Attribute Name	Total O...	Docum...
abl	abl	3	3
abou	abou	1	1
absolut	absolut	14	14
absorb	absorb	1	1
abysm	abysm	1	1
accept	accept	3	3
accid	accid	1	1
accommod	accommod	2	2
accost	accost	1	1
account	account	2	2
achiev	achiev	1	1
acquaint	acquaint	1	1
activ	activ	3	3
ad	ad	3	3
add	add	1	1



5. Edited the previous process by replacing the “Stem (Porter)” operator with the “Generate n-Grams” operator within the sub-process window of the “Process Documents” operator and saved the process as “05-n-grams refinement” respectively



Given the max length limitation of two word generation in the “Generate n-Grams (Terms)” operator, shown below are the results of the context in which the word “absolute” is attached to in order minimize the misinterpretation of the feedback which in this case can be concluded that the word “absolute” appears in generally positive reviews of the hotel

