UNIVESITY OF MILAN

DATA SCIENCE AND ECONOMICS

Algorithms for Massive Datasets

Course Project

Market-Basket Analysis (Finding Frequent Itemsets)

Tweets about the Russo-Ukrainian War

Student: Uladzislau Luksha 964000

E-mail: Uladzislau.luksha@studenti.unimi.it

Department of Economics, Management and Quantitative Methods

December 2022

I declare that this material, which I now submit for assessment, is entirely my own work and has not been taken from the work of others, save and to the extent that such work has been cited and acknowledged within the text of my work. I understand that plagiarism, collusion, and copying are grave and serious offences in the university and accept the penalties that would be imposed should I engage in plagiarism, collusion or copying. This assignment, or any part of it, has not been previously submitted by me or any other person for assessment on this or any other course of study.

Abstract

In this project, we studied FP-Growth technique to perform frequent itemsets mining on a large-scale data. The data used for the study was part of a dataset containing tweets about the Russo-Ukrainian war. The dataset was downloaded from the Kaggle repository. Given the size of the dataset and the importance for scaling up, Spark environment was used for the computations. Since the proposed project directly involves working with text, preliminary research and data pre-processing were carried out, including removing identical tweets, text cleaning, stop-words removal and tokenization. The analysis is performed for two languages: Ukrainian and English. After the implementation of FP-Growth algorithm for English tweets, we studied the scalability of the code, varying the size of the dataset and minimal support parameter.

1 Dataset

Ukraine Conflict Twitter Dataset [1] is released under the CC-BY-SA 4.0 license. It contains tweets about the Russo-Ukrainan war and is being updated on a daily basis starting from February 27, 2022. The current size of the dataset is 13.2 GB. Data on the tweets of each day of the war are presented as a separate CSV file. Because these files are quite big themselves, they are presented in condensed form. The dataset was downloaded directly to the Google Colab. For convenience of data processing, as well as for meaningful analysis of recent events, the period from December 6 to 10, 2022 was chosen as a studied sample. The dataset contains 27 features but for further analysis we will use only the "text" and "language" columns. The "text" column contains the text of the tweet itself, and the "language" column will be used to filter out the desired data. The number of rows of the studied dataset before the pre-processing is 279769.

2 Pre-processing

The first step was to filter out all tweets in English and Ukrainian. This choice was dictated by the fact that English is the most popular language for tweets, while Ukrainian is a direct participant in the conflict, and the author of the project speaks these languages. Further, since we are looking for

text	tokenized
icymi oil prices ...	[opec, cap, russia..
a global fertilis..	[new, invasion, b...
news n napproxima...	[forced, freezing...
entering at day 2...	[invasion, since,..

Table 1: Cleaned data. (Using English tweets as an example)

the most frequent itemsets, each tweet will be considered a basket, and each word in it will be an item. In the first step of cleaning up the text, we checked the missing values, removed all duplicates, all url links, the hashtag "#" symbol, and the mentioning "@"symbol. The second cleaning step was done individually for each language after creating a Spark data frame for them. Subsequent text processing includes removing punctuation, text tokenization, removing extra spaces and stop-words. To remove stop words in English, we used the nltk library [2], and for ukrainian we downloaded a separate set of stop words from the repository on GitHub. Emoji were also removed for English (for Ukrainian this step was skipped, because the encoding of symbols of the Cyrillic alphabet requires a different approach to cleaning emoji). Also, an important step in text processing was to clean up repetitive words in each tweet, since we need each basket to consist of unique elements. After all the preprocessing procedures, the data sample for analysis is as follows (Table 1).

Here the "tokenized" column can be used as input for the FP-Growth algorithm. FP-Growth requires itemsets to be in the format of a dataframe and the baskets to be an array of strings, which was successfully done.

3 Algorithms considered

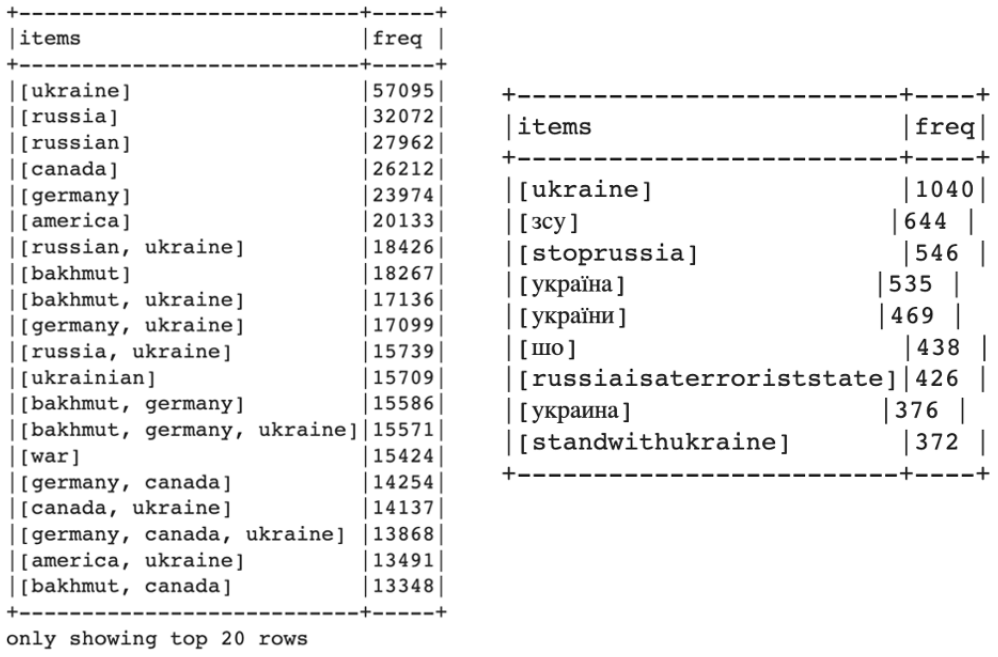
For the purposes of this analysis, the Frequent Pattern Growth (FP-Growth) algorithm was used. Initially, the idea was to compare its performance with Apriori algorithm. The problem with the latter is that it is unlikely to be scalable. This is caused by the fact that the candidate itemsets will always be huge for a dataset with a large number of frequent items or a low support value. In addition, the Apriori algorithm scans the dataset several times to determine the frequency of the itemsets. Because it incurs high computational costs, the Apriori algorithm is considered slow and will not be considered for further analysis [3].

On the contrary, FP-Growth is considered to be an effective and scalable tool for finding frequent items. The fundamental difference is in the organization of the data. FP-growth uses tree data structures, not sets, as Apriori does [4]. First, FP-Growth algorithm counts the occurrences of individual words and puts them in a table. Then, non-frequent words are removed using the minimum support

parameter: all the words with fewer occurrences are rejected. Those words that have not yet been rejected are sorted in order of occurrence. The last step is building a frequent pattern tree, adding the baskets on by one. Each word stands for a node in the tree and it comes with a counter that is used for finding frequent itemsets later. When a pattern is discovered, the relevant item's count is increased. Once this FP tree is built, this allows the data to be scanned faster.

4 FP-Growth implementation

After pre-processing all the text data, we searched for frequent itemsets, using FP-Growth algorithm. The threshold value of the support parameter for different languages was chosen by a brute-force method so as to increase the number of relevant results. The threshold support values were chosen as 0.05, 0.02 for English and Ukrainian, respectively. The decrease in the threshold for Ukrainian can be explained by the fact that the total number of tweets in these languages is significantly lower than in English.



items	freq
[ukraine]	57095
[russia]	32072
[russian]	27962
[canada]	26212
[germany]	23974
[america]	20133
[russian, ukraine]	18426
[bakhmut]	18267
[bakhmut, ukraine]	17136
[germany, ukraine]	17099
[russia, ukraine]	15739
[ukrainian]	15709
[bakhmut, germany]	15586
[bakhmut, germany, ukraine]	15571
[war]	15424
[germany, canada]	14254
[canada, ukraine]	14137
[germany, canada, ukraine]	13868
[america, ukraine]	13491
[bakhmut, canada]	13348

only showing top 20 rows

items	freq
[ukraine]	1040
[зсу]	644
[stoprussia]	546
[україна]	535
[україни]	469
[шо]	438
[russiaisaterroriststate]	426
[украина]	376
[standwithukraine]	372

Figure 1: Frequent items found by FP-Growth for English (left) and Ukrainian (right) data with minimum support thresholds 0.05, 0.02 respectively

If we translate a set of the most frequent Ukrainian words into English, we obtain the following words:

`[[ukraine], [armed forces of ukraine], [stoprussia], [ukraine], [ukraine], [what], [russiaisaterroriststate], [ukraine], [standwithukraine]]`

Several positions of "ukraine" in the most repeated Ukrainian words are caused by the fact that in the original language it is in different cases. The number of frequent items in the English dataset is 76, while after the first execution of the FP-growth algorithm, the number of frequent items in the Ukrainian dataset is only 9. The results of these two tests are fairly consistent. What is interesting about the English dataset is the emphasized presence of Canada, Germany, and the United States. And also, the city of Bakhmut. Nevertheless, based on the news agenda, these results are obvious, because after the massive battles in the city of Bakhmut this city was practically destroyed. In this regard, the U.S. announced the delivery of the Patriot missile defense system to Ukraine. And Germany and Canada announced new financial aid packages. To see how the most frequent items are related, we can look at the table of associative rules (Figure 2).

antecedent	consequent	confidence	lift	support
[bakhmut, america, germany, ukraine]	[canada]	0.9991123305358296	6.0195946340223925	0.07837365642447491
[germany, canada, russian, ukraine]	[bakhmut]	0.9619570644033949	8.318601121746328	0.06098471900439313
[germany, canada, russian, ukraine]	[america]	0.9282076884672991	7.289718189308134	0.05884512641320724
[bakhmut, ukraine]	[germany]	0.9085714285714286	5.987179863060229	0.09863648448478864
[america, germany, canada]	[bakhmut]	0.9629082426127528	8.326826511851708	0.07838631673566536
[america, germany, canada]	[ukraine]	0.9981337480559875	2.76058651765462	0.08125387722030207
[bakhmut, germany, canada]	[america]	0.9285392921415717	7.292322452735404	0.07838631673566536
[bakhmut, germany, canada]	[ukraine]	0.999850029994001 2	2.765333321164473	0.0844062947067239
[germany, canada, ukraine]	[bakhmut]	0.9614247602566875	8.313997978803917	0.0844062947067239
[germany, canada, ukraine]	[america]	0.9255173408320715	7.268589430803166	0.08125387722030207

only showing top 10 rows

Figure 2: Association rules (English dataset)

The English dataset association rules study only confirms the validity of the hypotheses about the reasons for the frequencies of Canada, the United States, and Germany.

However, one could have done less complicated analysis to find that the most recurring elements in the dataset about the Russo-Ukrainian war would be "russia," "ukraine," "war," "ukrainian," "putin," etc. Since these words do not reveal the valuable context of the period under study, it was decided to reanalyze the dataset using the FP-growth algorithm, having previously excluded the most popular words that do not provide the necessary context.

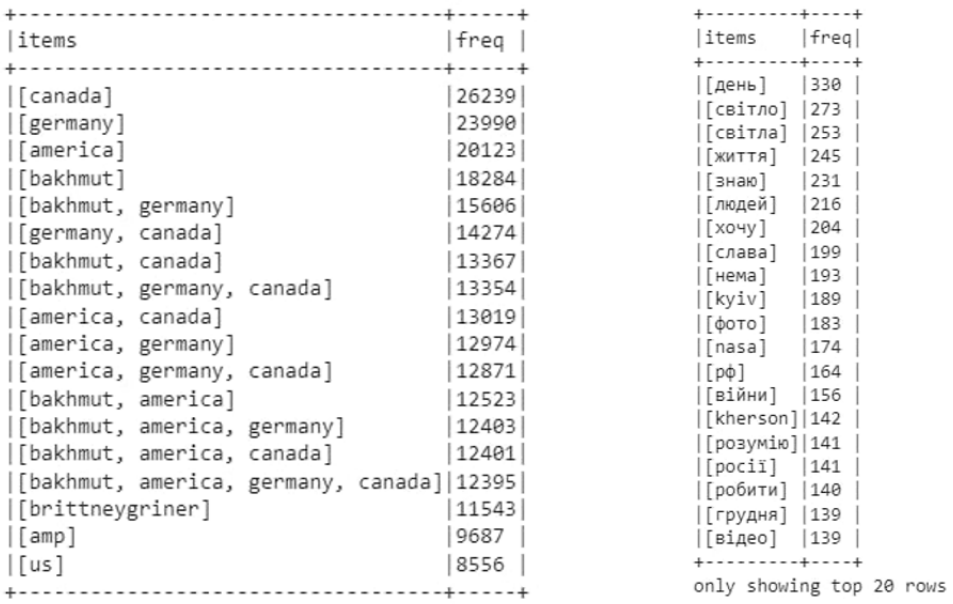
To do this, we created the following lists (frequentEN and frequentUA), then appended them to already existing lists of stop words and repeated the text cleaning procedure.

When we reanalyzed frequent items, we lowered the minimum support parameter for the Ukrainian dataset to 0.005. The results of the secondary analysis are shown in Figure 3.

While the results for the English dataset are almost unchanged, the frequent items of the Ukrainian dataset have undergone significant changes.

Translated into English, the results of the reanalysis of the Ukrainian dataset are as follows:

[[day],[light],[light], [life],[know],[people],[want],[glory],[none],[kyiv],[photo],



items	freq
[canada]	26239
[germany]	23990
[america]	20123
[bakhmut]	18284
[bakhmut, germany]	15606
[germany, canada]	14274
[bakhmut, canada]	13367
[bakhmut, germany, canada]	13354
[america, canada]	13019
[america, germany]	12974
[america, germany, canada]	12871
[bakhmut, america]	12523
[bakhmut, america, germany]	12403
[bakhmut, america, canada]	12401
[bakhmut, america, germany, canada]	12395
[brittneygriner]	11543
[amp]	9687
[us]	8556

items	freq
[день]	330
[світло]	273
[світла]	253
[життя]	245
[знаю]	231
[людей]	216
[хочу]	204
[слава]	199
[нема]	193
[kyiv]	189
[фото]	183
[nasa]	174
[рф]	164
[війни]	156
[kherson]	142
[розумію]	141
[росії]	141
[робити]	140
[грудня]	139
[відео]	139

only showing top 20 rows

Figure 3: Frequent items found by secondary computation of the FP-Growth algorithm (after additional stop-words removal) for English (left) and Ukrainian (right) data with minimum support thresholds 0.05, 0.005 respectively

`[nasa],[russian federation],[wars],[russia],[kherson],[understand],[december],[do],[video]]`

These results can also be interpreted. The bombing of Ukraine's critical energy infrastructure has consistently left the citizens of Ukraine living without light. Nasa video was also made public, showing the dire blackout situation in Ukraine. Thus, conducting a secondary analysis to search for frequent items, revealed elements of the news agenda previously hidden behind more popular and obvious words.

5 Scalability

In order to check the scalability of the proposed solution, we measure the computation time by changing the size of the input data for the FP-Growth algorithm being used. We will measure the computation time with respect to an English dataset that has undergone secondary text clearing. Moreover, before that we study the dependence of the computation time on the change of the minimum support parameter (Figure 4). Based on general logic, the smaller the chosen minimum support parameter, the more time it will take the algorithm to calculate frequent items. Nevertheless, for our dataset we do not observe such a pronounced trend in the decrease of computation time. However, we confirm that this logic is valid, because the environment crashed when we tried to calculate data

processing time with smaller values of the minimum support parameter.

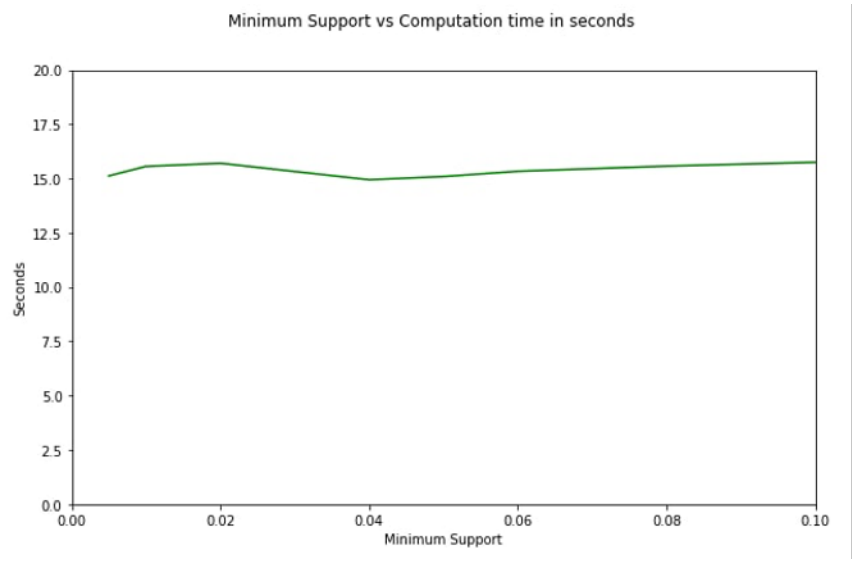


Figure 4: Computation time depending on the change of the minimum support parameter.

To study the scalability of the algorithm, we chose a minimum support parameter of 0.05. And also formed an array with the values of the dataset fractions from 0.1 to 1 at intervals of 10%. The scalability results of the algorithm are shown in Figure 5.

Dataset Fraction	Computation time
0.1	7.014691
0.2	7.907520
0.3	7.868286
0.4	8.324123
0.5	8.332984
0.6	8.644070
0.7	8.631411
0.8	9.050065
0.9	9.267596
1.0	9.713839

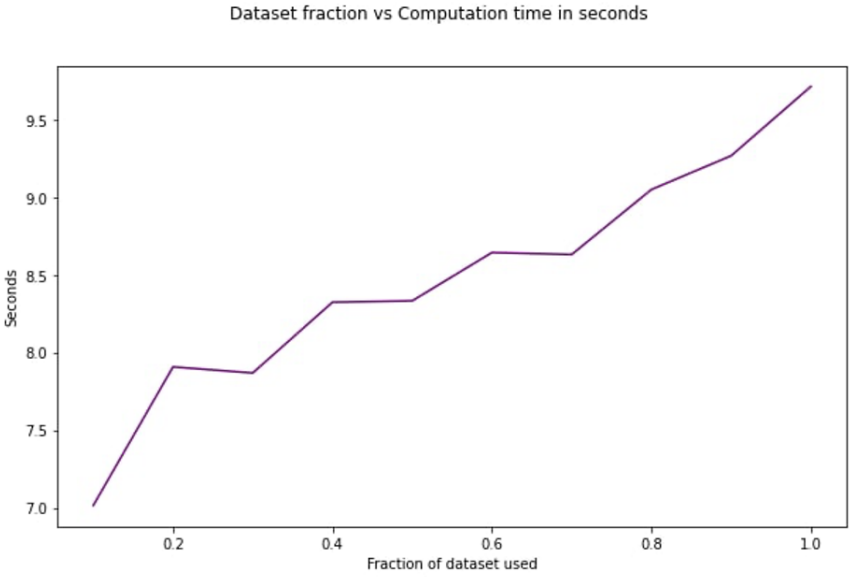


Figure 5: Computation time depending on the change of the dataset fraction..

Based on the results obtained, we can state that the proposed algorithm has a fairly good scalability, since all fractions of the dataset were analyzed without overloading RAM. In order to predict how the algorithm will behave on even larger data, we use polynomial regression, assuming that the

computation time will increase quite steadily with the size of the dataset.

At this point, the war in Ukraine has been going on for almost 10 months, or 40 weeks. Since in this subset we analyze only one week's worth of data in order to predict the computation time of English tweets over the entire dataset, we multiply the available size (157974 baskets) by 40, obtaining 6318960 baskets. The linear regression forecast of the computation time for such a value is around two minutes (1:46), while the forecast of a 3rd degree polynomial regression for such a value is 5 days and 11 hours.

6 Conclusion

Our experiment shows that PySpark implementation of FP-Growth is quite an efficient algorithm in terms of computation speed. It was able to detect relevant frequent items responsible for Ukraine's news agenda. To improve performance of the algorithm under study, we can create topic-related stop-word dictionaries in advance to help filter important frequent items more clearly, as well as focus on calculating pairs, triplets, and even larger frequent items, which will allow a more holistic understanding of the context of the events.

7 References

1. Kaggle Ukraine Conflict Twitter Dataset. *https://www.kaggle.com/datasets/bwandowando/ukraine-russian-crisis-twitter-dataset-1-2-m-rows*
2. Example using nltk for preprocessing text. *https://colab.research.google.com/github/gal-a/blog/blob/master/docs/notebooks/nlp/nltk_preprocess.ipynb#scrollTo=-44aMwUcQZxmFfff*
3. Frequent Pattern (FP) Growth Algorithm In Data Mining. *https://www.softwaretestinghelp.com/fp-growth-algorithm-data-mining/#Shortcomings_Of_Apriori_Algorithm*
4. FrequentPatternMining. *https://en.wikipedia.org/wiki/Frequent_pattern_discovery*
5. Apache Spark SQL Guide. *https://spark.apache.org/docs/latest/sql-getting-started.html*