Algorithm for Massive Dataset: Frequent Skills in LinkedIn Jobs 2024

Hiyab Negga

December 2024

Abstract

The purpose of this project highlights the use of Apriori algorithm in finding association with positive interest from the skills that we find in job listings. Our analysis was based on a sample of LinkedIn 2024 containing a total of about 1.3 million jobs. We relied on Spark to run our algorithm on our sample and found that the most frequent skills listed are predominantly soft skills. This coincides with the reality as most of the white collar jobs require soft skills regardless of the industry. We used a 10% as a threshold and observed teamwork → communication), (leadership → communication) and (customer service → communication) have confidence greater than 0.5 along with a positive interest.

"I/We declare that this material, which I/We now submit for assessment, is entirely my/our 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/our work, and including any code produced using generative AI systems. I/We 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/us or any other person for assessment on this or any other course of study."

Contents

1 Introduction 3

2 Data Pre-processing 3

3 Methodology 4

3.1 Algorithm 4

3.2 Spark 5

4 Exploratory Data Analysis 6

5 Results & Discussion 7

6 Conclusion 8

1 Introduction

Gaining a clear overview of the job market is crucial for job seekers, educational institutions, employers, and recruiters. Education institutions can equip their students with the skills sought out by employers by helping them prepare when they enter the job market. Furthermore, job seekers can understand what is required by employers so that they can sharpen their skills to meet the standards demanded by employers. On the other hand, employers can benefit from the insight in helping them better define the skills combination they seek. In this regard, the multiple stakeholders can best serve by assessing the combination of skills that frequently occur across job listings to gain an understanding and best position themselves by knowing the common skill sets in demand.

In this short project, we will leverage a data mining technique that would enable us to utilize 'Market Basket Analysis' to help us identify the most frequent skill sets in job listings. We will be utilizing LinkedIn Jobs & Skills data set collected in 2024. The focus of the project is on the job_skills comma-separated data frame consisting of two columns, namely, job_link and the job_skills, and 1,296,381 rows.

Our approach will utilize PySpark to analyze the most frequent skill set within the data to ensure scalability and computational efficiency in processing. We will begin by cleaning data and pre-processing our job_skill data. Subsequently, we will briefly explore the exploratory data analysis and assess the most frequent skill sought after to gain a short synopsis before applying the Apriori Algorithm. Before discussing the results of our analysis we need to gain an understanding of the algorithm.

2 Data Pre-processing

Before conducting our analysis we need to pre-process the data and have it in a format that would be suitable to conduct our algorithm for the market basket analysis. The first action was to assess if we had any null values or any duplicates in our data. Our data contained 2007 null values for the 'job_skills' column and no duplicates were found. Since we already have an ample amount of data we removed the null values.

Secondly, when inspecting the data we found that the job listings were from different countries and thus different languages were used for the 'job_skill' column. To handle the different discrepancies we mainly relied on regular expressions (regex), where we kept alphanumeric characters, replaced 'and', 'or' and '&' with a comma because different skills could be represented as 'communication and customer service' where we would treat the conjunction as two separate skills. In addition, we removed stop words and appended the words 'skill' and 'skills' so that we can focus more on the skill listed. Special characters # and discarded if they were used as a listing but they were kept in order to accommodate cases where C or C++ are listed as skills.

Finally, to keep the preprocessing data uniform we trimmed the white space and transform them to lower case. Now that we have all the skills properly processed we can use word cloud to visualize and our data is be ready to apply the Apriori algorithm.

3 Methodology

3.1 Algorithm

With the aim of the project focused on applying data mining techniques, specifically the market-based model that would enable us to find frequent skill set. With the model gaining popularity in the retail industry to find and commonly is described by using the terminologies item and baskets. In this case items which consist of the individual items that are included when purchasing products whether it be in a physical store or an e-commerce platform. A basket on the other hand is a set of items (itemset) sometimes referred to as "transaction". In order to keep things within our context we will be referring to items as skill and we will use listing to refer to listing.

The foundation of the discovering frequent skills set will be based on the A-Priori algorithm. To best understand let us define some concepts, namely the support and the confidence.

$$\text{Support(S)} = \frac{\text{Freq(S)}}{N} \tag{1}$$

Where:

- Freq(S) is the frequency of the skill in the listings
- N represents the total number of listings

If we were considering the support of two skills then their support can be represented as the union or the frequency of both of the items occurring together:

$$\text{Support}(\text{S}_1, \text{S}_2) = \frac{\text{Freq}(\text{S}_1 \cup \text{S}_2)}{N} \tag{2}$$

Where:

- Freq($\text{S}_1 \cup S_2$) is the frequency where both skills occur

$$\text{Confidence}(\text{S}_1 \rightarrow \text{S}_2) = \frac{\text{Support}(\text{S}_1 \cup \text{S}_2)}{\text{Support}(\text{S}_1)} \tag{3}$$

The confidence is the conditional probability on observing the two skills S_1 and S_2 together. We can concisely represent the confidence as Confidence($\text{S}_1 \rightarrow \text{S}_2$) = $\mathbb{P}$ (S_2 | S_1).

The Apriori algorithm applies a breadth-first strategy. It start with the unique skills which are referred to as 1-skill(k = 1) sets within the data set. If the item satisfies the pre-specified minimum support threshold then it would include it in the next iteration (k = 1) to form a larger skills set.

The iterative process is repeated and the combinations with the lowest probability are pruned until no more frequent skills sets are found. Result of our algorithm give us a comprehensive associations of the skills set. We finalize by assessing the confidence of all of the skill sets and keep the associations that have a confidence grater than pre-specified threshold.

Using the confidence, which assess the probability that skills are listed together will enable us to extract the association rule that is commonly expressed as {IF} → {THEN}. The {IF} component is known as the antecedent and the {THEN} component is known as the consequent. This enables us to predict the probability that a certain skills may be listed together in job listings.

But we might have regularly bought items that might distort our insight to determining association. The concept of interest becomes important to evaluate the associations that are significant. We do this by assessing if the confidence is different from the occurrence of the skill. We basically do this by subtracting the the confidence from the support of the consequent.

$$\text{Interest} = \text{Confidence}(\text{S}_1 \rightarrow \text{S}_2) - \text{Support } (\text{S}_2) \tag{4}$$

Where :

- Interest = 0, S_1 has no influence on S_2
- Interest > 0, Listing that contain S_1 tend to also contain S_2
- Interest < 0, Listing that contain S_1 tend not to contain S_2

But it is important to note that the multi-iteration process is computationally expensive and memory intensive because for each iteration the Apriori algorithm scans the database to compute supports of k-skills sets. In this project we will be focusing on k = 3.

3.2 Spark

To implement our project we have resorted to the Spark framework. The main advantage of using the Spark framework in Python (PySpark) is to utilize the Resilient Distributed Dataset (RDD). This is an abstraction methods enable us to enhance the performance for our relatively large dataset. RDD represents a collection of items distributed across many compute node that can be manipulated in parallel.[2]

We have utilized the MapReduce programming style to process each of the job listing in finding the frequent skills. We pruned out the less frequent skills and used the results of the support to conduct further analysis of the association and interest for which we didn't have to rely on the MapReduce.

In the first pass of the algorithm we use the transform each of the skills in a listing and aggregate the values by the key, in this case the skills to which we then filter the skills that fall below our

[2] H, Karau, et.al, Learning Spark: Lightning-Fast Big Data Analysis

threshold.

flatMap → reduceByKey → filter → singeltons

Then between each pass we need to generate the candidates from the singletons we have produced to form a 2-skill set. We used a simple nested for loop where we use a union to form pairs of skills. To ensure there are no duplicates we made sure we only kept distinct pairs. Then the generated candidates would be broadcast to all of the nodes we can quickly and efficiently use them when we check for subsets.[3]

In the second and third passes the process is simimar where we take the generated candidated and do a count of the skills that have been identified and procude to prune the skills that fall below the threshold.

flatMap → reduceByKey → filter → doubleton/tirpeltons...

4 Exploratory Data Analysis

To get an idea of the frequency of the different skills that are listed in our data we have resorted to WordCloud visualization. In Figure 4.1, we can see skills such as communication, customer service, teamwork, leadership, problem solving and time management appear more frequent than other skills. The visualization helps in a way what we would be expecting to see when we apply our Apriori algorithm.

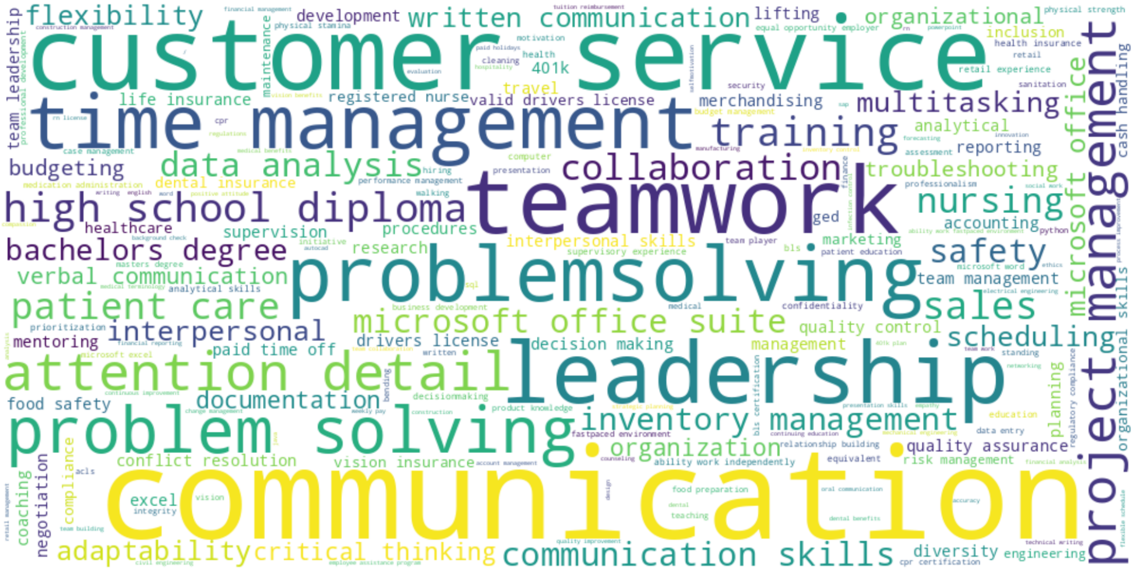

Figure 4.1: WordCloud of Skills

[3] P. Singh, et.al, A Data Structure Perspective to the RDD-based Apriori Algorithm on Spark

5 Results & Discussion

Since we had a large data we sampled 10% of the listings and ran the Apriori algorithm. Results from the analysis indicate that the most frequent skills that are listed are predominantly soft skills. The table below shows the support reported in relative frequency of the most frequent skills with 3-iteration of generating candidates with support threshold set to 10%.

Skill	**Support**
communication	0.38
customer service	0.22
leadership	0.15
customer service, communication	0.13
teamwork, communication	0.13
problemsolving	0.12
time management	0.12
problem solving	0.12
communication, leadership	0.11
attention to detail	0.11

Table 5.1: Results of Frequent Skills where s = 10%

Skills such communication, customer service and leadership have a relatively higher support when compared to other frequent singletons. Where as we can see that customer service, communication as well as teamwork, communication and communication, leadership are the most frequent doubletons. However, even though we generated 3 candidates and examined their support there were none that were above the threshold. After assessing the results of the different skills we can now assess the confidence and the interest of the doubletons that appear in our table.

Antecedent	**Consequent**	**Confidence**	**Interest**
teamwork	communication	0.72	0.33
leadership	communication	0.68	0.30
customer service	communication	0.62	0.23
communication	customer service	0.36	0.13
communication	teamwork	0.34	0.15
communication	leadership	0.28	0.12

Table 5.2: Results of Confidence and Interest

Based on the results we can observe that if we set a threshold of confidence of above 0.5 the results that would be important and have a relatively higher interest are (teamwork → communication), (leadership → communication) and (customer service → communication). The interest for the these association is also greater than zero, thus we would expected that listing that contain teamwork will also tend to contain communication. The same reason applies for leadership and communication, respectively. A possible reason for this could be that since these are the soft skills that are highly sought for in addition to the technical abilities within any industry.

6 Conclusion

In this project we assessed the job listing that were collected from LinkedIn for the year of 2024 in the hopes of identifying the frequent skills. We started by inspecting the data and cleaned all of the discrepancies that were observed such as encountering listings in different languages, symbols used as a way of listing skills, removed stop words along with the words 'skill' and 'skills'. In order to get an idea of the most frequent skills we used word cloud to get a quick idea of what we should be expecting for our analysis.

To run the Apriori algorithm on a sample data we relied on PySpark by fundamentally applying the MapReduce programming style to handle large amounts of data over the RDD. By doing this we were able to set a relatively high support (s = 10%) and generated 11 frequent skills consisting of singletons and doubletons. To finalize the analysis we computed the confidence and the interest of the doubletons. Results from the analysis indicate that (teamwork → communication), (leadership → communication) and (customer service → communication) have confidence greater than 0.5 along with a positive interest.

However it is important to note that some other interesting associations could be extracted from the sample by lowering the number support threshold to lower than 10% and also increasing the candidate set to generate. It should also be noted we have only carried our the analysis only on a small fraction and the algorithm could be scaled in order considering a more efficient variation of the Apriori algorithm such as the PCY would likely to better results. Also adding a layer of hashing or applying triangular matrix could have a significant impact on performance.