Applied Machine Learning in Local Supermarket

Yejin Kim
School of Management Engineering
UNIST
kimyejin99@unist.ac.kr

Jiyeong Min
School of Management Engineering
UNIST
jymin1999@unist.ac.kr

Abstract

This project aims to identify the most effective rules and patterns and recommend products by adopting various data mining techniques such as TF-IDF, association rules, and recommendation systems. Also, by providing effective rules and patterns, it will be helpful for supermarkets to retain loyal customers or VIPs. In general, consumers do not just buy one item when they buy a product at a supermarket, but rather buy several types of items at the same time. Therefore, if we can find the correlation between the various types of products that consumers purchase, we can use this to prepare a useful sales strategy. In addition, if retailers know the composition of customers' shopping baskets, they can use them to prepare products or make product arrangement decisions.

I. Introduction

Today, supermarkets can immediately accumulate information about consumers' purchasing behavior through POS, so they can make decisions about product preparation and sales using many data collected. However, these large data consist of shopping basket information about hundreds of foods. Because it is not formalized, it is difficult to identify consumers' purchasing behavior by applying existing demand analysis methodologies. Recently, data mining techniques have drawn attention to analyze large amounts of data and extract new information.

Association Rules Analysis, one of these data mining techniques, provides useful information about the association between items from customers' purchasing information. This gives sellers insight into who their customers are and why they have such a buying pattern, what products they buy together, and which products are effective for promotion. The results of the analysis will also be useful when retail stores develop bundled products, form shelves, or create product promotional flyers to be sent to consumers.

This project aims to identify the most effective rules and patterns and recommend products by adopting various data mining techniques such as TF-IDF, association rules, and recommendation systems. Specifically, this study intends to conduct under the following four goals.

Goal 1: Classify product names according to item type using TF-IDF

Goal 2: Discover the association rules between purchased products.

Goal 3: Improve product placement based on the relationship between products.

Goal 4: Predict future purchases and establish a product recommendation system.

II. Literature Review

2.1. Association Rules Analysis

Association Rules Analysis (Agrawal et al. 1993; Agrawal & Srikant 1994) is a technique that finds the relationship between data by using the frequency and probability of co-occurrence and expresses it as a rule. It is being used in various fields (김남규 2008; 안현철 외 2006; 윤성준 2005; Burke 2000; Wang et al. 2004; Wang et al. 2007) such as shopping basket analysis, Internet shopping mall recommendation system, cross-selling, store layout, catalog design, and promotion strategy establishment.

2.2. NLP (Natural Language Processing)

The type of the data varies widely. Previously, the focus was on how to deal with structured data that occurs a lot in the system, but now the spread of SNS is turning on the unstructured data growth. Unstructured data is not only textual conversation, but also photos and videos that is large-scale and complex in everyday life. Gartner, a U.S. information technology research, and advisory firm said that more than 80% of the rapidly growing data in the company is unstructured data rather than structured. Among unstructured data, the importance of text data is also increasing in analysing customers' needs, such as taste, sensibility, and preference. (Chowdhury et al., 2003)

Korean Natural Language Processing (KoNLP), which we will use, is a study of text mining using Korean. Most of them utilize analysis methods such as information extraction, document classification, and clusters based on frequency analysis. Semantic analysis studies are being conducted using mining techniques such as TF-IDF, bag of words (BOW), N-gram, non-negative matrix factorization (NMF), and Word2Vec. Most of the research is conducted with English-based, and there is a lack of Korean-language-based research. Korean-based words are constantly being created with abbreviated expressions of netizens freely expressing emotions or forms. So, it is more meaningful for researchers to process Korean-based natural language.

Since each language has its own characteristics for natural language processing, using natural language processing tools suited for English is not suitable for Hangeul. For this reason, KoNLPy, which is customized for Korean natural language processing, appeared. KoNLPy is a Python package for natural language processing of Korean language. Using this package, we can do the Morphological analysis. Morphological analysis is the identification of the structure of morphemes and other linguistic units, such as root word, affixes, or parts of speech. POS(Part-of-speech) tagging is the process of marking up morphemes in a phrase, based on their definitions and contexts. In KoNLPy,

there are several different options you can choose for POS tagging. All have the same input-output structure; the input is a phrase, and the output is a list of tagged morphemes. (Jeon et al., 2016; Lee et al., 2016)

2.3. TF-IDF

Word is constructed by the sentence, and we utilize the frequency of the word in sentence to numerically present the association of the document for each word. TF (Term frequency) is a value that indicates how often a particular word appears in a document. The higher the value, the more important it is in the document, under the hypothesis that if a word appeared several times, it would have higher relevant in that document.

$$TF = tf(t,d) \quad \dots (1)$$

The equation (1) is the frequency of the word t in document d. However, there may be errors in considering if the frequency of words is high, the relation with sentence also high. For example, in Korean, there are words that are essential to build a sentence, such as "을/를" and "이/가" but do not mean much in context. These are called 'stop words', which can be perceived as important words because they appear frequently in text. However, it is not a substantially important word and should be removed in advance.

IDF is being used to correct these errors in TF. Some words often appear even though they are not related to a sentence. The TF-IDF technique is utilized as a technique to limit these unrelated words. TF-IDF is a text data algorithm used by search engines. When there is a research document or sentence, statistical figures indicate how important a particular word is within the research document. It is also utilized in search engines to determine similarities between documents or ranking search results

$$\text{IDF} = \log\frac{D}{1+df(t)} \quad \dots (2)$$

This equation (2) expresses the reverse of the frequency of document. You can divide the total number of sentences by the number of sentences in which the word appeared. That is, how common a word is throughout a document. After dividing the total number of documents by the number of documents containing the word, take the log.

$$TF-IDF = tf(t,d) * log\frac{D}{1+df(t)} \quad \dots (3)$$

In equation (3), the definition of TF-IDF is the product of word frequency and inverse document frequency, and when there are multiple documents, it shows how important a particular word is within a particular document. As such, TF-IDF is a numerical representation of the association of a sentence for each word. Most researchers express many results in English-based TF-IDF analysis. The absence of Korean-based research is interpreted as a delay in analysing the needs of the people. (Yun-tao et al., 2005)

2.4. Recommendation system

The recommendation system is now an important part of the Fourth Industrial Age. Starting with content recommendations such as images or videos, there are no places where recommendations are not included. It can be used in all fields, such as shopping, exposure to searching, to AD and SNS. Previously, customer segmentation was required when applying marketing to a product. However, the recommendation system allows marketing tailored to individual. The recommendation is, after all, a prediction. Recommendations are made after predicting the outcome that the user would prefer. Therefore, the purpose of the recommendation system is to recommend suitable items for the user.

Look at the Fig. 1. Recommendation systems are largely divided into content-based filtering and collaborative filtering. In addition, collaborative filtering methods are again divided into K-Nearest neighbour and Latent Factor methods. Content-based filtering is a method of recommending other songs with similar content if the user likes a particular song. For example, if a user likes a snack from a certain brand, they recommend another snack from the same brand that has similar characteristics. Nearest neighbour collaborative filtering can be divided into user-based and item-based. User-based is a method of recommending similar values by measuring associations based on users and item-based is based on items. In general, item-based methods are more accurate than user-based ones. Finally, latent factor collaborative filtering is a technique that allows us to make recommendation predictions by extracting latent factors hidden in the user-item rating matrix. These potential factors are difficult to define specifically but play a major role in laying the groundwork for actual recommendations. (Pazzani et al., 2007; Zhao et al., 2010)

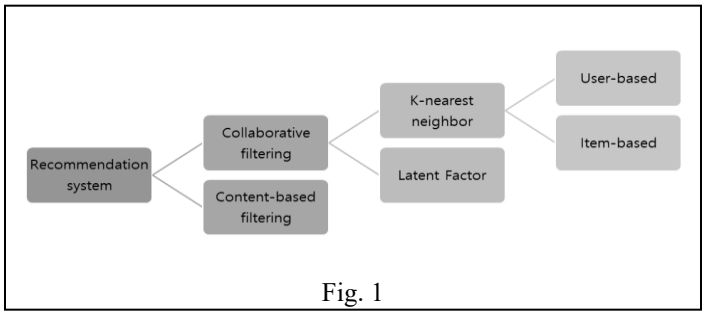

Fig. 1

Surprise is a python scikit for building and analysing recommendation systems. It provides various ready-to-use prediction algorithms such as neighbourhood methods, matrix factorization-based (SVD, SVD++). It also provides various similarity measures (cosine, MSE, Pearson). Using this package, we can evaluate, analysis and compare the algorithms' performance.

Using the built-in data, we can make the simple recommendation system. But if you want to implement this package in your data, it needs to modify the data using Reader () class. This class have to be like 'user; item; rating; [timestamp]'. (Hug et al., 2020)

III. Method and Application

3.1. Data description

Today, supermarkets can immediately accumulate the data of consumer and their purchasing behavior through POS. Through POS machine, seller can manage products and tables, order and make payments, and quickly identify or register products by scanning bar codes. Of course, the company that provides pos provides basic services such as sales flow, but it is not up to the point of obtaining meaningful results from customer data and delivering them to the seller. In this paper, we

will find meaningful information through the sales data and apply the data into various machine learning method to use it for supermarket operation.

Where did all the local supermarket that used to play the role of asking each other how they were and playing with friends at the arcade in front of the stationery store go? Recently, franchises, online shopping and early morning delivery have become popular, and local supermarket are disappearing. While the number of convenience stores increased by 1,600, the number of local supermarkets decreased by 1,169. According to the Seoul Metropolitan Government, the business environment of local supermarket is getting worse. This is due to the increase in franchise stores and the expansion of online shopping. Average monthly sales at local supermarket in the commercial district are 17 million won. It is 3 million won less than the average monthly sales of 20 million won per store in Seoul.

According to a survey on consumption trends of Seoul citizens last year, purchases through large companies, franchises, and online are increasing. Seoul citizens purchase 48% of daily necessities at large supermarkets, SSM, franchises, and convenience stores. Then, 23% of online shopping and 19% of others such as department stores. The place of purchase of daily necessities such as toilet paper and detergents, and miscellaneous goods such as stationery, hardware, and kitchenware, which used to be purchased in the local supermarket, is changing to large marts and online. In the case of food, the proportion of purchases in local supermarket districts is relatively high, but there is a possibility that even this will be taken away due to a surge in home meal service. Only 25said they would use local supermarket when purchasing food. (뉴시스, dongA)

Major supermarkets are captivating customers by optimizing and reorganizing their space through customer data. Through data analysis, they have developed strategies and led to sales improvement. However, in the case of a local supermarket, there is no way to use these data even if they are gathered in a pos machine. So, we thought about the ways to improve sales and competitiveness through data analysis. (박해욱, 서울경제; 박민주, 서울경제)

To get the data, we tried to contact with many supermarkets, and as a result, we were able to get data from one local supermarket. It is located in an apartment complexes, elementary schools and hospitals. Nearby, there are lots of competitors. From here, we were able to get about 1,000 sales data and each data looks like Fig. 2. It consists of trade name, barcode number, unit price, the number of purchasing, real price and point.

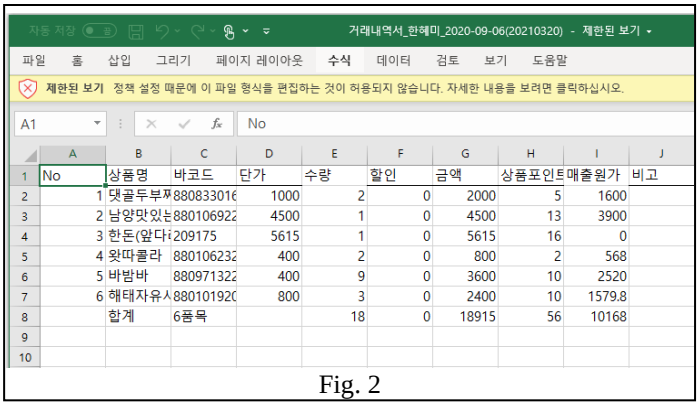

Fig. 2

3.2. Product extraction & classification

The first thing what we do using data was name extraction and classification. The data frame that we had was looked like Fig. 3.

	No	상품명	바코드	단가	수량	할인	금액	상품포인트	매출원가	비고
0	1	댓골두부찌개용300g	8808330161130	1000	2	0	2000	5	1600	
1	1	남양맛있는우유GT기획	8801069221397	4500	1	0	4500	13	3900	
2	1	한돈(앞다리살)찌개용	209175	5615	1	0	5615	16	0	
3	1	왓따콜라	8801062321773	400	2	0	800	2	568	
4	1	바밤바	8809713220055	400	9	0	3600	10	2520	
...	...	...	...	...	...	...	...	...	...	...
3767	975	농심쌀새우깡80g	8801043036054	1150	1	0	1150	3	850	
3768	975	농심양파깡77g	8801043036498	1300	1	0	1300	3	965	
3769	975	농심매운새우깡90g	8801043036078	1150	1	0	1150	3	850	
3770	975	재사용봉투10리터	9144539122350	310	1	0	310	0	285	
3771	975	밀크플러스900ml*2	8806371306381	2980	1	0	2980	13	2500	

Fig. 3

The value of '상품명' column is complex. There were total of 1,359 such complexly written product names and there are no spacing words in each value. Therefore, the process of changing to simpler was needed. For example, the product name "댓골두부찌개용300g" should be changed as "두부", and "남양맛있는우유gt기획" should be changed as "우유".

It's too time consuming to manually work on all data. So, we used TF-IDF, which is used in NLP. As mentioned above, TF-IDF is a statistical figure of how important a word is within a particular document. So, the more a particular word appears in a document, and the less it appears in all documents, the more important it becomes. However, in the case of algorithms we have to use, the more likely the word is used in many documents, the more representable. For example, so many product names are grouped under a keyword called "두부". Therefore, we used a modified tf-df algorithm to give more weight to the appearance of many words in the many documents.

$$modified\ TF-DF = Term\ Frequency * Document\ Frequency$$

So, we first separated our data into morphemes using the Konlpy package. Secondly, we applied the modified tf-df algorithm to each data. Finally, we extracted the word with the largest weight. As a result, important words were selected from the original name.

But there was a problem. That is, this algorithm is useless in the product name that doesn't contain the keyword. In the case of 왓따콜라, 바밤바, 해태자유시간, they should be included in 껌, 아이스크림, 제과류 respectively, but they are not. In such cases, the name of item was often included in the brand name, so we used this fact to bind the product. So, we grouped the number of 1,359 products into 264 upper versions. Look at the Fig. 4.

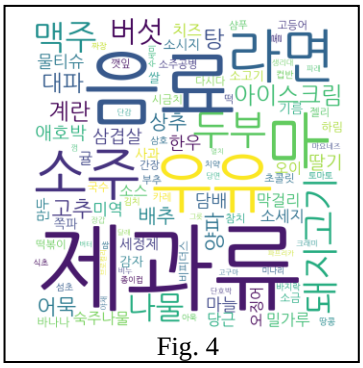

Fig. 4

3.3. Association Rules Analysis

Association rule analysis is one of the data mining techniques, and the results can be used for sales strategies such as store display, bundled product development, and cross-selling. Association rule analysis is a technique first introduced by Agrawal et al. (1993). It aims to identify new rules by analyzing associations between entities included in an event or transaction. Shmueli et al. (2009) named it "Market basket analysis" because the associated rule analysis applied in the field of marketing is similar to the form of finding associations between products purchased together in a shopping basket purchased by a customer.

The association rule is expressed in the form of 'X→Y', which means 'many people who buy product X purchase product Y'. As for indicators for evaluating the derived association rules, there are support, confidence, and lift (Paul & David, 2015).

1) Support

It refers to the ratio of product X and product Y simultaneously transacted among all transactions as shown in equation (4). The usefulness of the rule can be grasped through support.

Support(X→Y)

$$= Pr(X \cap Y) = \frac{Number\ of\ transactions\ involving\ products\ X\ and\ Y\ at\ the\ same\ time}{Total\ number\ of\ transactions} \quad \ldots (4)$$

2) Confidence

It refers to the ratio in which product X and product Y appear simultaneously among transactions that purchase product X, as shown in equation (5). This is expressed in the form of a conditional probability when product Y appears during a transaction involving product X, indicating the strength of the association rules.

Confidence(X→Y)

$$= \frac{Pr(X \cap Y)}{Pr(X)} = \frac{Number\ of\ transactions\ involving\ products\ X\ and\ Y\ at\ the\ same\ time}{Number\ of\ transactions\ with\ product\ X} \quad \ldots (5)$$

3) Lift

It is the ratio of product X and product Y traded at the same time divided by the ratio of product X and product Y respectively traded, which is equal to the value obtained by dividing the confidence by the probability of product Y appearing. That is, it refers to the ratio between the case of arbitrarily purchasing product Y and the case of purchasing considering the relationship with product X, and Lift(X→Y) has the same result value as Lift(Y→X). The lift indicates whether the derived association rule is coincidental or actually has an association. Since the lift is not a probability value, it can have a value greater than 1. If the lift between two products is greater than 1, it means that the two products have a positive correlation with each other. The lift is calculated as in equation (6).

Lift(X→Y)

$$= \frac{Pr(X \cap Y)}{Pr(X) \times Pr(Y)} = \frac{Confidence}{Pr(Y)} = \frac{Number\ of\ transactions\ involving\ products\ X\ and\ Y\ at\ the\ same\ time}{\frac{Number\ of\ transactions\ with\ product\ X}{Total\ number\ of\ transactions} \times \frac{}{Number\ of\ transactions\ with\ product\ Y}} \quad \ldots (6)$$

3.4. Apriori algorithm

In association rule mining, since the number of subsets excluding the empty set increases exponentially as 2k – 1 (k is the number of items.) in the set of all items that appear in the transaction, the computation of support for all cases is enormous. So, we use Apriori principle to reduce the amount of computation. In other words, the strategy to reduce the number of all possible item-sets (M) is the Apriori algorithm. An item set with support greater than or equal to the minimum support is called a frequent item-set. The main content of the Apriori algorithm is to calculate the association rule by finding only the frequent item-sets above the minimum support, instead of calculating the support for all item-sets. Apriori principle is that if an item-set is frequent, then all of its subsets must also be frequent. For this, anti-monotonicity, which is a characteristic of support, is used as the calculation method. It means that the support of an item-set never exceeds the support of its subsets.

Depending on the value of confidence and support, the content of associated rules may vary. Since there are no specific thresholds for confidence and support, in this project, except for the condition that the lift should be greater than 1, we arbitrarily adjust the thresholds of confidence and support by identifying the content or number of associated rules that are expressed.

3.5. Application of Apriori algorithm

3.5.1. Convert the data frame into a list.

To use the association analysis function, the data frame must be converted into a list. Product names are grouped according to the transaction number. Look at the Fig.5. Product names are grouped according to the transaction number. And we remove the empty list and duplicate values as there are cases where multiple purchases of the same product are made in one transaction. Fig. 6 is the final sales list according to the transaction number.

```
data_arr=[[]for i in range(976)]
num=0

for i in data['최종상품명'] :
    data_arr[data['No'].iloc[num]].append(i)
    num+=1
data_arr.pop(0)
num=0

for i in data_arr :
    data_arr[num] = list(set(data_arr[num]))
    num+=1

data_arr
```

Fig. 5

```
[['우유', '두부', '음료', '제과류', '돼지고기', '아이스크림'],
 ['우유', '두부', '음료', '소시지', '버섯', '돼지고기'],
 ['라면', '우유', '무', '제과류', '아이스크림'],
 ['딸기', '아이스크림'],
 ['애호박', '우유', '두부', '음료', '무', '식초'],
 ['우유', '제과류', '음료', '아이스크림'],
 ['계란', '우유', '제과류', '탕'],
 ['우유', '물티슈', '당근', '무', '탕', '조명'],
 ['비피더스', '제과류', '애호박', '밀가루'],
 ['라면', '우유', '버섯', '무', '제과류', '탕', '비피더스'],
 ['음료', '귤'],
 ['우유', '물티슈', '음료', '떡', '제과류', '만두', '탕'],
 ['우유', '제과류'],
 ['부탄가스'],
 ['부추', '물티슈', '만두'],
 ['양파', '딸기', '두부', '오일세럼', '돼지고기', '소스'],
 ['비피더스', '우유', '마요네즈'],
 ['라면', '깻잎', '배추'],
 ['바나나'],
```

Fig. 6

3.5.2. Transform the list using "mlxtend".

The reason for this transformation is to apply the Apriori algorithm. *mlxtend* is a Python library that supports statistical analysis.

In the Fig. 7, the dataset has a unique label through the fit function, and the transform function converts the Python list into a NumPy array. For all data, indicate True if it exists in each list and False if it does not.

```
from mlxtend.preprocessing import TransactionEncoder
from mlxtend.frequent_patterns import apriori, association_rules

te = TransactionEncoder()
te_ary = te.fit(data_arr).transform(data_arr)
df = pd.DataFrame(te_ary, columns=te.columns_)
df
```

	가지	간장	간청각	갈비	감자	감자전분	강남콩	건포도	계란	고구마	...	햇반	행주	호박	호박	홍어	홍합	황태	후추	헤이설음	휴지
0	False	False	False	False	False	False	False	False	False	False	...	False	False	False	False	False	False	False	False	False	False
1	False	False	False	False	False	False	False	False	False	False	...	False	False	False	False	False	False	False	False	False	False
2	False	False	False	False	False	False	False	False	False	False	...	False	False	False	False	False	False	False	False	False	False
3	False	False	False	False	False	False	False	False	False	False	...	False	False	False	False	False	False	False	False	False	False
4	False	False	False	False	False	False	False	False	False	False	...	False	False	False	False	False	False	False	False	False	False
...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...
970	False	False	False	False	False	False	False	False	False	False	...	False	False	False	False	False	False	False	False	False	False
971	False	False	False	False	False	False	False	False	False	False	...	False	False	False	False	False	False	False	False	False	False
972	False	False	False	False	False	False	False	False	False	False	...	False	False	False	False	False	False	False	False	False	False
973	False	False	False	False	False	False	False	False	False	False	...	False	False	False	False	False	False	False	False	False	False
974	False	False	False	False	False	False	False	False	False	False	...	False	False	False	False	False	False	False	False	False	False

Fig. 7

3.5.3. Apply the Apriori algorithm.

Using Apriori, we can build the frequent item sets. And using the previous table, we will extract data with 'support' of 0.02 or higher. Here, depending on the value of support and confidence, the content of association rules may vary. So, we arbitrarily adjust the thresholds of support and confidence by identifying the content and number of association rules that are expressed. Look at the Fig. 8. If you look at the case of two item sets, you can see that the two items are related to each other. In particular, confectionery and drink have the highest support value, and milk and confectionery have the next highest value.

```
frequent_itemsets = apriori(df, min_support=0.02, use_colnames=True)
frequent_itemsets
```

	support	itemsets
0	0.053333	(계란)
1	0.047179	(고추)
2	0.025641	(귤)
3	0.063590	(나물)
4	0.033846	(당배)
5	0.027692	(당근)
6	0.056410	(대파)
7	0.075897	(돼지고기)
8	0.107692	(두부)
9	0.035897	(딸기)
10	0.093333	(라면)
11	0.031795	(마늘)
12	0.028718	(막걸리)
13	0.082051	(맥주)
14	0.144615	(무)
15	0.030769	(물티슈)
16	0.029744	(미역)
17	0.026667	(밀가루)
18	0.044103	(배추)
19	0.053333	(버섯)
20	0.020513	(사과)
21	0.029744	(삼겹살)
22	0.040000	(상추)
23	0.025641	(소세지)
24	0.023590	(소스)
25	0.111795	(소주)
26	0.026667	(숙주나물)
27	0.048205	(아이스크림)
28	0.036923	(애호박)
29	0.047179	(양파)
30	0.048205	(어묵)
31	0.024615	(오이)
32	0.023590	(오징어)
33	0.153846	(우유)
34	0.217436	(음료)
35	0.232821	(제과류)
36	0.026667	(치즈)
37	0.052308	(탕)
38	0.028718	(한우)
39	0.023590	(라면, 음료)
40	0.026667	(라면, 제과류)
41	0.023590	(맥주, 소주)
42	0.025641	(맥주, 음료)
43	0.023590	(맥주, 제과류)
44	0.023590	(소주, 무)
45	0.024615	(무, 음료)
46	0.032821	(무, 제과류)
47	0.020513	(소주, 우유)
48	0.027692	(소주, 음료)
49	0.024615	(소주, 제과류)
50	0.022564	(제과류, 아이스크림)
51	0.031795	(우유, 음료)
52	0.036923	(우유, 제과류)
53	0.061538	(제과류, 음료)

Fig. 8

3.5.4. Generate the rules.

Using the association rules function, we can find out which item sets have a positive correlation among item sets with support over 0.02. Fig. 9 is an association rule with a lift of at least 1. So, we can see that the following items are positively correlated with each other.

```
network_df=association_rules(frequent_itemsets, metric="lift", min_threshold=1)
network_df
```

	antecedents	consequents	antecedent support	consequent support	support	confidence	lift	leverage	conviction
0	(라면)	(음료)	0.093333	0.217436	0.023590	0.252747	1.162399	0.003296	1.047255
1	(음료)	(라면)	0.217436	0.093333	0.023590	0.108491	1.162399	0.003296	1.017002
2	(라면)	(제과류)	0.093333	0.232821	0.026667	0.285714	1.227187	0.004937	1.074051
3	(제과류)	(라면)	0.232821	0.093333	0.026667	0.114537	1.227187	0.004937	1.023947
4	(맥주)	(소주)	0.082051	0.111795	0.023590	0.287500	2.571674	0.014417	1.246604
5	(소주)	(맥주)	0.111795	0.082051	0.023590	0.211009	2.571674	0.014417	1.163447
6	(맥주)	(음료)	0.082051	0.217436	0.025641	0.312500	1.437205	0.007800	1.138275
7	(음료)	(맥주)	0.217436	0.082051	0.025641	0.117925	1.437205	0.007800	1.040669
8	(맥주)	(제과류)	0.082051	0.232821	0.023590	0.287500	1.234857	0.004487	1.076743
9	(제과류)	(맥주)	0.232821	0.082051	0.023590	0.101322	1.234857	0.004487	1.021443
10	(소주)	(무)	0.111795	0.144615	0.023590	0.211009	1.459106	0.007422	1.084150
11	(무)	(소주)	0.144615	0.111795	0.023590	0.163121	1.459106	0.007422	1.061330
12	(소주)	(우유)	0.111795	0.153846	0.020513	0.183486	1.192661	0.003314	1.036301
13	(우유)	(소주)	0.153846	0.111795	0.020513	0.133333	1.192661	0.003314	1.024852
14	(소주)	(음료)	0.111795	0.217436	0.027692	0.247706	1.139216	0.003384	1.040238
15	(음료)	(소주)	0.217436	0.111795	0.027692	0.127358	1.139216	0.003384	1.017835
16	(제과류)	(아이스크림)	0.232821	0.048205	0.022564	0.096916	2.010498	0.011341	1.053939
17	(아이스크림)	(제과류)	0.048205	0.232821	0.022564	0.468085	2.010498	0.011341	1.442297
18	(우유)	(제과류)	0.153846	0.232821	0.036923	0.240000	1.030837	0.001105	1.009447
19	(제과류)	(우유)	0.232821	0.153846	0.036923	0.158590	1.030837	0.001105	1.005638
20	(제과류)	(음료)	0.232821	0.217436	0.061538	0.264317	1.215610	0.010915	1.063725
21	(음료)	(제과류)	0.217436	0.232821	0.061538	0.283019	1.215610	0.010915	1.070013

Fig. 9

3.5.5. Filter the rules.

We filter the rules to make more reliable rules. Fig. 10 is an association rule with a lift of at least 1 and confidence of at least 0.2. the lift is highest for beer and soju. The confidence is highest for ice cream and confectionery. The support is highest for drink and confectionery. So, completed final association rule is Fig, 11. Here, look at the third and last rules. Beer and Soju and Confectionery and Drink have double arrow. So, these are two-way association rules.

```
network_df[(network_df['lift'] >= 1) & (network_df['confidence'] >= 0.2) ]
```

	antecedents	consequents	antecedent support	consequent support	support	confidence	lift	leverage	conviction
0	(라면)	(음료)	0.093333	0.217436	0.023590	0.252747	1.162399	0.003296	1.047255
2	(라면)	(제과류)	0.093333	0.232821	0.026667	0.285714	1.227187	0.004937	1.074051
4	(소주)	(맥주)	0.111795	0.082051	0.023590	0.211009	2.571674	0.014417	1.163447
5	(맥주)	(소주)	0.082051	0.111795	0.023590	0.287500	2.571674	0.014417	1.246604
7	(맥주)	(음료)	0.082051	0.217436	0.025641	0.312500	1.437205	0.007800	1.138275
9	(맥주)	(제과류)	0.082051	0.232821	0.023590	0.287500	1.234857	0.004487	1.076743
10	(소주)	(무)	0.111795	0.144615	0.023590	0.211009	1.459106	0.007422	1.084150
14	(소주)	(음료)	0.111795	0.217436	0.027692	0.247706	1.139216	0.003384	1.040238
16	(아이스크림)	(제과류)	0.048205	0.232821	0.022564	0.468085	2.010498	0.011341	1.442297
18	(우유)	(제과류)	0.153846	0.232821	0.036923	0.240000	1.030837	0.001105	1.009447
20	(음료)	(제과류)	0.217436	0.232821	0.061538	0.283019	1.215610	0.010915	1.070013
21	(제과류)	(음료)	0.232821	0.217436	0.061538	0.264317	1.215610	0.010915	1.063725

Fig. 10

Ramen → Drink

Ramen → Confectionery

Beer ⇄ Soju

Beer → Drink

Beer → Confectionery

Soju → Radish

Soju → Drink

Ice cream → Confectionery

Milk → Snack

Confectionery ⇄ Drink

Fig. 11

3.5.6. Top 15 Selling Products

From now on, let's analyze the best-selling products. Fig. 12 shows the Top 15 Selling Products. We can see that confectionery, drink, milk, radish, and Ramen were sold the most in that order.

```
x = data["최종상품명"].value_counts()
x = x.sort_values(ascending = False)
x = x[:15]

ditribution_plot(x=x.index, y=x.values, yaxis="Count", xaxis="Products")
```

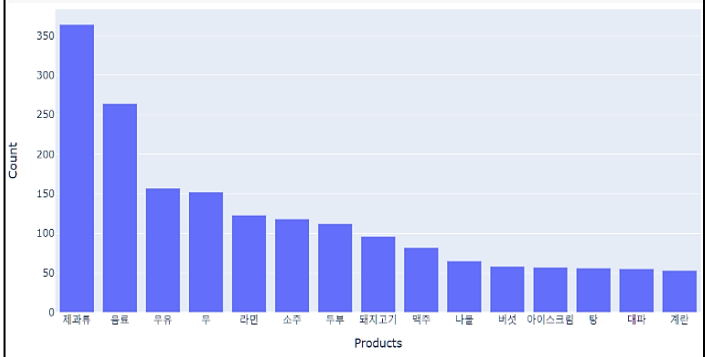

Fig. 12

3.5.7. Network Visualization using "networkx"

We made a network visualization using the *networkx* module in Python. Based on the total sales of each product, a circle is created, and it serves as a 'node' of the network graph. And based on the results of the association analysis, a line is created, and it serves as a 'relationship' of the network graph. So, it is a network graph created based on the association analysis results and the total sales of each product which we previously checked. In this way, we can see the relationship between products directly. Look at the Fig. 13. And the confectionery with the highest sales is located in the center and has a deep connection with other products.

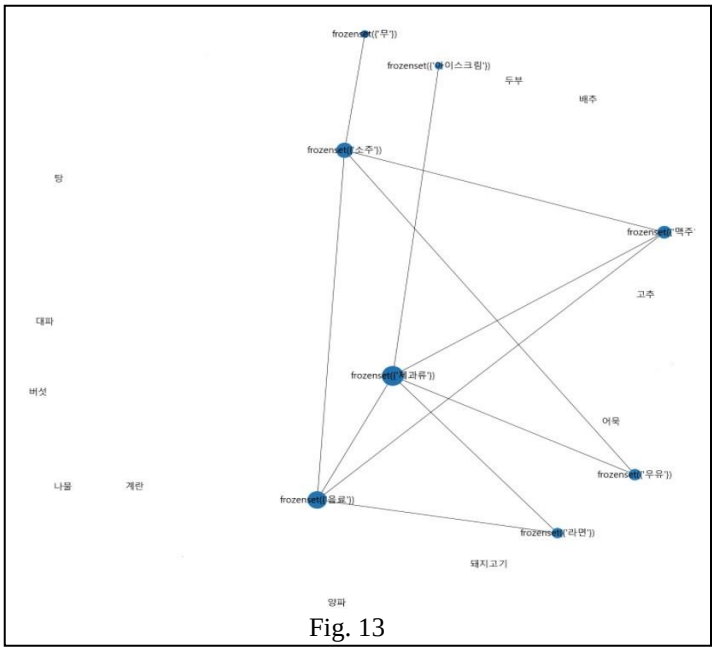

Fig. 13

3.6. Recommendation

The last thing what we do was the recommendation system. A recommendation system is a recommendation of a product suitable for an individual based on user preferences and past behaviors. when you think up the scene according to local supermarket, the retailer greets customers with pleasure and say, 'this product is fresh today'. This could be the charm of a local supermarket. However, this does not reflect individual's tendency or past behavior, but it simply contains preferences of retailer. Moreover, it is not sustainable because it is only in that position. So, we used the recommendation system and thought about how to provide customized recommendation services so that customers can continue to stop by this local supermarket.

To use the recommendation system algorithm, we used a surprise package. We used user, product name, and the number of purchases as columns. Collaborative filtering has a variety of algorithms. Among them, we applied the methods of als, sgd, and knn and proceeded with cross validation, respectively. The results showed that rmse value of als is 1.10, sgd is 1.11, and knn is 1.20. So, we chose the als algorithm and do parameter tuning.

After that, we made the system of recommendation for a specific user. Look at the Fig. 14. If we make a service that recommends to user 1, we run a recommendation algorithm based on the historical data and predict the expected number of purchases. So, if a new item comes, you can recommend the item to the user that have high probability of purchasing.

mean_test_mae	std_test_mae	rank_test_mae	mean_fit_time	std_fit_time	mean_test_time	std_test_time	params	param_bsl_options	param_k
0.446746	0.027134	11	0.016296	0.000465	0.034929	0.000845	{'bsl_options': {'method': 'als', 'reg': 1}, ...	{'method': 'als', 'reg': 1}	2
0.437805	0.027972	5	0.015498	0.000409	0.036686	0.000291	{'bsl_options': {'method': 'als', 'reg': 1}, ...	{'method': 'als', 'reg': 1}	3
0.437538	0.029731	3	0.016364	0.000572	0.037462	0.001363	{'bsl_options': {'method': 'als', 'reg': 1}, ...	{'method': 'als', 'reg': 1}	4
0.434426	0.030408	1	0.015795	0.000438	0.038035	0.000671	{'bsl_options': {'method': 'als', 'reg': 1}, ...	{'method': 'als', 'reg': 1}	5
0.446746	0.027134	12	0.015470	0.000401	0.034581	0.000476	{'bsl_options': {'method': 'als', 'reg': 2}, ...	{'method': 'als', 'reg': 2}	2
0.437805	0.027972	6	0.016161	0.000614	0.035929	0.000749	{'bsl_options': {'method': 'als', 'reg': 2}, ...	{'method': 'als', 'reg': 2}	3

물 를 6.381479692214732 개 살 것으로 예상됩니다.
식품 를 5.243953088459811 개 살 것으로 예상됩니다.
속옷 를 4.422806698022963 개 살 것으로 예상됩니다.
귤 를 3.173309571337812 개 살 것으로 예상됩니다.
참치 를 2.7846524359282867 개 살 것으로 예상됩니다.
비트 를 2.7355514809268993 개 살 것으로 예상됩니다.
굴 를 2.6081415462868778 개 살 것으로 예상됩니다.
누룽지 를 2.5357365305955932 개 살 것으로 예상됩니다.
스티커 를 2.508117562623645 개 살 것으로 예상됩니다.

Fig. 14

IV. Results and Discussion

4.1. Association Rules Analysis

It shows the sales strategy based on the analysis results so far. First of all, there are four sales strategies.

1. Focus on the best-selling products and prepare enough stock for these items.

2. Display the products at 130-140cm on the floor where the eyes are focused.

3. Display the best-selling products in the back of the store to attract customers to every corner of the store.

4. Display the products that are difficult to manage inventory such as fruits and vegetables near the entrance and increase the sales rate by attracting interest.

Also, we improved the product placement. Look at the Fig. 15. The best-selling confectionery was placed in the center, and correlated products were placed close together.

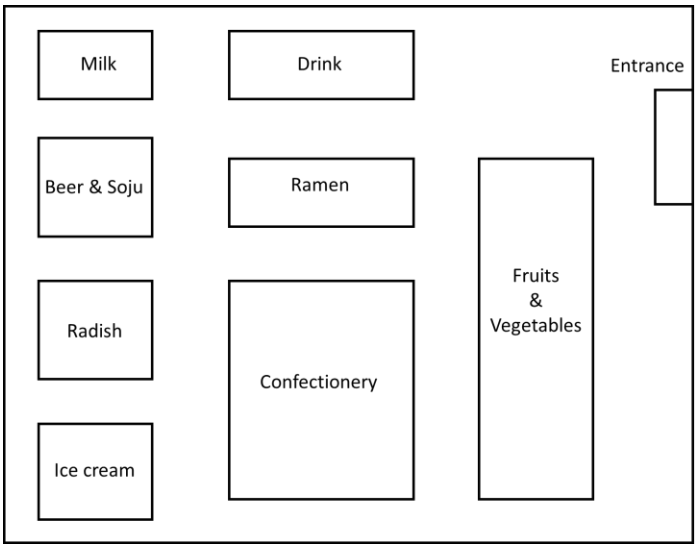

Fig. 15

4.2. Recommendation

Look at the Fig. 16. The results of the recommendation system can be utilized as follows. First, when an item arrives in the supermarket, the boss enters the item in the POS system. Then, the program runs a recommendation algorithm based on past customer data. If a new item is likely to be preferred by a customer, each customer will receive the recommendation message. It is good for each customer to be notified about the product they want to buy, and the boss can induce the customer to act on the next purchase, which leads to making profits.

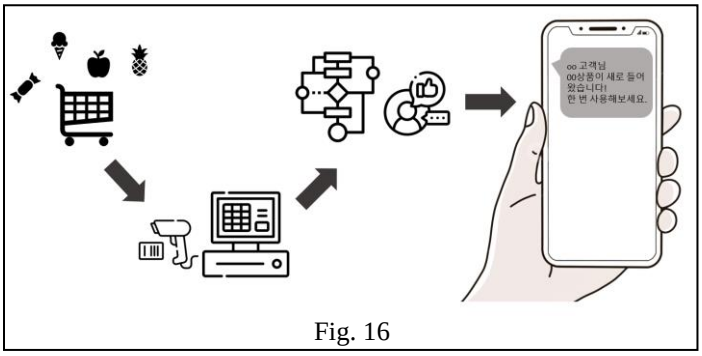

Fig. 16

V. Conclusion

This project was conducted to identify the most effective rules and patterns and recommend products. First, we classified product names according to item type using TF-IDF. Second, we discovered the association rules between purchased products. Third, we improved product placement based on the relationship between products. Fourth, we predicted future purchases and

establish a product recommendation system. By providing effective rules and patterns, it will be helpful for supermarkets. It will also be useful when developing bundled products, forming shelves, or creating product promotional flyers to be sent to consumers. Moreover, supermarkets can recommend suitable items for the user by recommendation system.

REFERENCES

[1] Agrawal, R., Imielinski, T. and Swami, A. (1993), "Mining association Rules between Sets of Items in Large Databases," in Proc. ACM SIGMOD International Conference on Management of Data, Washington D.C., pp. 207-216, 1993.

[2] Agrawal, R. and Srikant, R. (1994), "Fast Algorithms for Mining Association Rules," International Conference on Very Large Data Bases, Santiago, Chile, pp.487-499, 1994.

[3] Burke. R (2000), "Knowledge-based recommender systems," Encyclopedia of Library and Information Systems, Vol. 69, 2000

[4] Chowdhury, Gobinda G. "Natural language processing." Annual review of information science and technology 37.1 (2003): 51-89.

[5] Hug, Nicolas. "Surprise: A python library for recommender systems." Journal of Open-Source Software 5.52 (2020): 2174.

[6] Jeon, Heewon, and Taekyung Kim. "Package 'KoNLP'." (2016).

[7] Lee, Jong-Hwa, and Hyun-Kyu Lee. "Research on Natural Language Processing Package using Open-Source Software." The Journal of Information Systems 25.4 (2016): 121-139.

[8] Paul A and David M. 2015. Data mining for the social sciences: An introduction. University of California Press.

[9] Pazzani, Michael J., and Daniel Billsus. "Content-based recommendation systems." The adaptive web. Springer, Berlin, Heidelberg, 2007. 325-341.

[10] Shmueli GN, Patel R and Bruce P. 2009. Data mining for business intelligence. Wiley

[11] Wang, W. F., Chung, Y. L., Hsu, M. H. and Keh, A. C. (2004), "A Personalized Recommender System for the Cosmetic Business," Expert Systems with Applications, Vol. 26, No. 3, pp. 427-434, 2004.

[12] Wang, W. F., Chung, Y. L., Hus, M. H. and Keh, A. C. (2007), "A Personalized Recommender System for the Cosmetic Business," Expert Systems with Applications, Vol. 26, No. 3, pp. 427-434, 2007

[13] Yun-tao, Zhang, Gong Ling, and Wang Yong-cheng. "An improved TF-IDF approach for text classification." Journal of Zhejiang University-Science A 6.1 (2005): 49-55.

[14] Zhao, Zhi-Dan, and Ming-Sheng Shang. "User-based collaborative-filtering recommendation algorithms on hadoop." 2010 third international conference on knowledge discovery and data mining. IEEE, 2010.

[15] 김남규(2008), "장바구니 크기가 연관규칙 척도의 정확성에 미치는 영향," 경영정보학연구, 제18권, 제2호, pp. 95-114, 2008.

[16] 뉴시스, "서울 동네가게의 몰락…편의점 1천6백개↑vs 슈퍼 1천개↓" , dongA

[17] 박해욱, "동네슈퍼가 살아남는 법, '진열대만 바꿔도 매출 쑥쑥', 서울경제.

[18] 박민주, "대형마트 매출 전략? 데이터에 물어봐" , 서울경제.

[19] 안현철, 한인구, 김경재(2006), "연관규칙기법과 분류모형을 결합한 상품추천시스템: G인터넷 쇼핑몰의 사례," Information System Review, 제8권, 제1호, pp. 181-201, 2006

[20] 윤성준(2005), "데이터 마이닝 기법을 통한 백화점의 고객 이탈예측 모형 연구," 한국마케팅저널, 제6권, 제4호, pp. 45-72, 2005