Published by :
http://www.ijert.org

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 7 Issue 06, June-2018

A Machine Learning Algorithm for Product Classification based on Unstructured Text Description

Vijay Nair
Director of Analytics Retail Company
Bengaluru, India

Supti Kanta Mohapatra
Alumnus of ISB Indian School of Business
Hyderabad, India

Reema Malhotra
Advanced Analytics Consultant Management Consulting
Company Gurugram, India

Nivedith Maknoor
Data Scientist Consultant Management Services Company
Hyderabad, India

Abstract - **Meticulous product classification is a key to acquiring new customers and drive customer retention for all online retailers. In this dynamic world, millions of products are added, removed and altered through multiple separate channels on a website and hence automation of this process has become the need of the hour. This paper describes the case study of one such solution that had been developed as a proof of concept for one of UK's leading retailer. Most solutions in the market classify products based on a proper text description, they can't identify products where the description is poor or incorrect. The algorithm described in this paper achieved this massive next-level business demand making it a much more advanced solution. In addition to being Client agnostic and portable across platforms, the overall approach of this solution is fairly simple yet profound, at the heart of which lies the engine to classify products based on unstructured text description. Thus, at a broader level, this approach is also industry agnostic and can be leveraged in other text mining problems as well.**

I. INTRODUCTION

The standard method of product ingestion involves manual intervention. See Figures section (Fig. 1. Manual Categorization Process)

In absence of an automated recommendation system, relying on human judgment is bound to decimate uniformity in product classification. The reason online retailers would focus on getting the product classification right across all levels is that this does not only impact where a product sits when a customer navigates through the website but even the search engine that recommends similar products to a customer also relies heavily on this hierarchical classification. Imagine searching for a laptop on a website and it recommends you a mobile.

Incorrect classification of products is destructive to a Retailer's reputation. In most cases, the misclassification of products results from human intervention during product Ingestion and classification apart from different channels of ingestion across separate business units. Let us consider a customer searching for a laptop of model x from brand y to discuss on these issues.

1) A data sourcing employee, P1 might place the laptop model X as Electronics → Computers & Accessories → Laptops → 14" Laptops → Laptop Model X

   While another employee P2 might think the classification Electronics → Computers & Accessories → Laptops → Brand Y → Laptop Model X as more appropriate.
2) Hierarchical classification errors are notorious as they have a tendency to increase exponentially. Assuming that the laptop model x was placed wrongly in the first level of a multi hierarchical classification structure, the error in the first level would cascade to subsequent levels.
3) Inappropriate classification could potentially result in a recommendation engine nightmare wherein a customer is suggested a different product altogether.

The algorithm discussed in this paper successfully mitigates the above-mentioned challenges in addition to accurately predicting classes for new products ingested in the system.

II. METHODOLOGY

A. Data Pre- Processing:

Different datasets were procured for each product type, though the same algorithm yielded amazing results on all of these, here is a short description of the datasets received:

Document Term Matrix generation: Post cleansing the data, a document term matrix is generated to obtain uni-gram and bi-gram tokens. In this problem, columns of a DTM would be the unique token (words and bi-grams) across all product descriptions in the training dataset (corpus), each product description (row) would represent a document in the corpus.

Dataset Name	No. of Rows	No. of Columns
Bakery	8467	53
Men's Apparel	58643	100
Ladies' Apparel	296296	15
Food& Grocery	55792	17

TABLE I DATASET SUMMARY

B. *Feature Selection Methods:*

a. Since the number of columns in a DTM are the number of unique tokens(words) in the entire training datasets, dimension reduction becomes extremely important before introducing classifiers.

b. Relying on simple Term Frequency (TF) is not advisable for classification problems like these as thresholding based on TF subtracts the rare tokens but in reality, occurrence of these rare tokens in a document might play a crucial role for tagging a product class. Two approaches explored in this algorithm are TFIDF and Chi-square tests:

i. TFIDF- this is a measure of how important a token is for a given document in the corpus, so if a token occurs more frequently in a particular document but less frequently across the corpus then that token gets a high score for the given document as it gives more information about that document.

TFIDF for a token t in a document d is computed as:

$$TFIDF(d,t) = TF(t) * IDF(d,t)$$

where,

$$IDF(d,t) = log[\frac{n}{DF(d,t)}] + 1$$

And n represents the number of documents in the corpus

The table below shows the top 10 words with highest TF-IDF scores for bakery products:

Feature	Score
Strong	1254
pack	531
wheat	445
flour	399
chocolate	371
cake	341
Milk	334
white	301
bread	289
bakery	274

TABLE II TF-IDF SCORES FOR BAKERY PRODUCTS

ii. Chi-square test- This is used to test the independence of two events. In this context it tests if the occurrence of a specific token in the document and the occurrence of the product class are independent or not. Let e_t = 1 is the document contains token t else e_t = 0, e_c = 1 if the document is in class c else e_c = 0, then the chi-square metric for the term t in document D for class c is given as:

$$X^2(D,t,c) = \sum_{e_t \varepsilon \{0,1\}} \sum_{e_c \varepsilon \{0,1\}} \frac{(N_{e_t e_c} - E_{e_t e_c})^2}{E_{e_t e_c}}$$

Where, N is the observed frequency in D and E is the expected frequency.

The table below shows the top 10 words with highest Chi-sq scores for bakery products:

Feature	Score
Euphorium	365
cake	344
free	191
sponge	138
rolls	129
Sliced	124
gluten free	122
Kingsmill	122
project	121
bloomer	117

TABLE III CHI-SQ SCORES FOR BAKERY PRODUCTS

The intuitive difference between TF-IDF and Chi-sq features is well established with the two tables, while TF-IDF weighs how important the token is for a given document, Chi-sq test also weighs the importance of the token in conjunction to the product classification, thus words like chocolate, strong, etc. would be significant to a product description but they may not be as crucial in defining the product classification as the words cake, gluten-free. Notice, most words appearing in the Chi-sq table describe the product attribute more than the literal product.

There are other techniques such as Information gain and mutual information which can be explored, although TFIDF and Chi- square yielded sufficiently accurate results for this algorithm.

C. *Classifiers:*

Once the feature selection is set, the next step is to choose the right classifier. The algorithm developed runs different classifiers on the same feature space and selects the final model based on accuracy results. Some of the classifiers that were considered for this algorithm are -SVM (Support Vector Machine), Naive Bayes Classifier, Random Forest.

a. SVM finds a hyperplane h, which separates different classes in the training corpus with the maximum margin.

Support vectors are those training examples which have the minimum distance from h. Since SVM assumes that all features are relevant and DTMs are sparse matrices, its extremely crucial to provide an appropriate underlying feature space for SVM to yield accurate results. This is the reason that for this algorithm SVM over a chi-sq DTM yielded best results for most datasets.

b. Naive Bayes uses a probabilistic model of text and works under strong assumptions. Word based unigram models assumes that words occur independently of other words in the document. The task is to estimate the probability that a document d with feature vector x_d is in the class C, i.e. P (C/x_d) with the perfect knowledge of

P (C^0/x_d) where C' denotes all classes except C. These assumptions are not ideal for a real case scenario which is why this classifier yielded weaker results in comparison to other classifiers.

c. Random forest operates by constructing multiple

decision trees at training time where final output is a class which is the mode of classes predicted from individual trees. Huge dimensionality of textual data increases the risk of excessive detailing while building the decision trees which leads to over fitting. Thus, even for Random Forests to work on a text classification problem, the underlying feature selection becomes extremely important. It was observed that both Random Forests and SVM are amongst the best learning classifiers for this text classification problem.

D. *Accuracy Measures:*

Instead of taking a single pair of train and test data, a K-fold cross validation is a much more robust technique where the advantage is that all observations are used for both training and validation and each observation is used for validation exactly once.

a. The algorithm selected the best classifier based on F- score which can be interpreted as a weighted harmonic mean of recall and precision. Intuitively, precision is the ability of the classifier not to label as positive a sample that is negative, and recall is the ability of the classifier to find all the positive samples.

E. *Assessment of the solution so far:*

a. While a usual machine learning algorithm for classification would stop at this stage, during the continued discussion with the business it was concluded that generating the product classes would help in classifying new products but this wouldn't add value to present classification which needed a standardization.

b. Analyzing the results across different categories helped in identifying another concern. It was observed that the same algorithm yielded magnificent results for categories like apparels and grocery but poor results for bakery. Further inspection revealed that the text description for apparels and grocery products which are mostly bought online was fairly rich in content but for products like bakery which are bought extensively from physical stores, the description was moderate and for some population of products rather poor.

c. This also highlighted that for some products the predicted classes were correct and uniform but original tagging which was manual was inconsistent and incorrect for some products.

d. These revelations magnified the requirement of an automated solution which could identify the products that needed better descriptions and the products whose tagging probably required a manual review along with highlighting the products which have been tagged automatically with significant confidence over the prediction.

VALIDATIONS & RESULTS:

Instead of only predicting the classes for each product, the algorithm could segment products into three categories-

"products that can be automatically classified", **"products that required a manual review"** and **"products with poor descriptive attributes"**.

$$F1 = \frac{2 * TP}{2 * TP + FP + FN}$$

where TP,FP,FN are numbers of true positives, false positives, and false negatives, respectively.

Level 2 F-Score		
Classifiers/Feature Space	TFIDF	ChiSq DTM
Random Forest	86.65 %	86.35 %
SVM	82.22 %	82.22 %
Nave Bayes	82.22 %	76.37 %

TABLE IV ILLUSTRATION OF THE ALGORITHMS OUTPUT FOR A GIVEN DATASET AND LEVEL OF HIERARCHY

b. This step is done iteratively over all levels in the hierarchy, each successive iteration adds the class predicted in previous level.

Levels	Bakery	Men's Apparel	Ladies' Apparel	Food & Groceries
Level 2	74 %	100 %	100 %	75.11 %
Level 3	62.6 %	100 %	100 %	76.26 %
Level 4	61.9 %	99.99 %	92.6 %	60.14 %
Level 5	36.8 %	99.84 %	70.6 %	50.31 %

TABLE V ACCURACY AT EACH LEVEL FOR DIFFERENT DATASETS

. This segmentation was done on the basis of a threshold as depicted below:

Illustration- Consider that the score for the prediction of a product was 0.9, this indicates that the algorithm is extremely sure of the prediction and hence the predicted class is fairly accurate, this product can be **classified automatically** .

If the score is 0.6 it means the algorithm is not sure about the prediction and hence this **product requires a manual review**.

If the score is 0.4, this reflects that the product description is poor, such products should be sent back to commercial buyers and ensured a rich content is provided.

Thus, based on this score, each product was classified Into of the above three segments.

See Figures section (Fig. 3. Product Classification into three segments)

In order to decide the right value of this threshold, a sensitivity report was also generated through the algorithm for each level which is represented in Fig 1.

See Figures section (Fig. 4. Level 2 Sensitivity Report)

The business could now choose the right threshold value based on the cost estimation to review certain product classifications manually or retrieving better descriptions.

This solution turned out to be extremely beneficial for the business as they now have a machine learning algorithm Which could validate, predict product classifications and identify products which required better descriptions.

CONCLUSION

Although the aim was to address the problem of a specific firm, the model discussed in this paper proved to be generic and portable in nature. With the use of various techniques like data consolidation, feature engineering, classifier selection, metrics for validation and finally segmentation of products that are automatable and otherwise, this solution has succeeded in minimizing human error and inappropriate classifications. It gives the flexibility to the business in order to strike the right balance in the trade-off between the cost of manual intervention and accuracy of prediction.

FIGURES

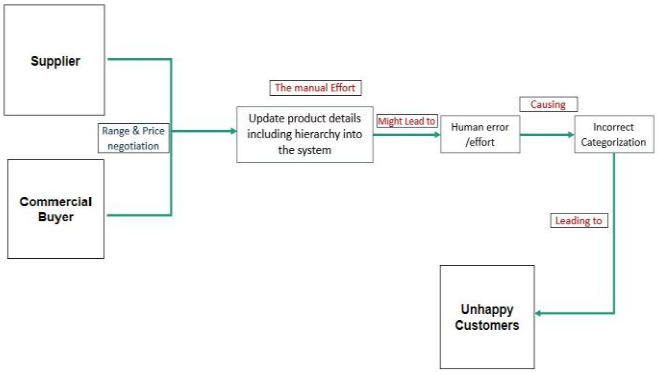


Fig. 1. Manual Categorization Process

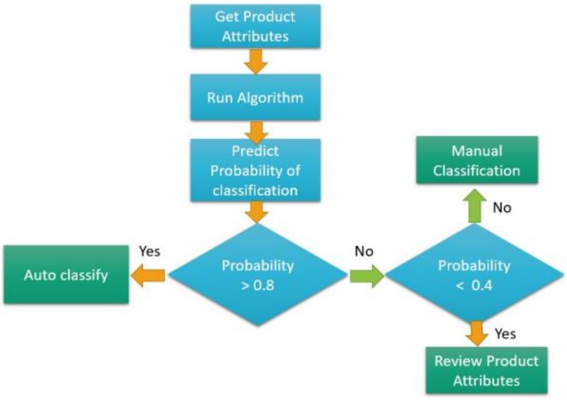
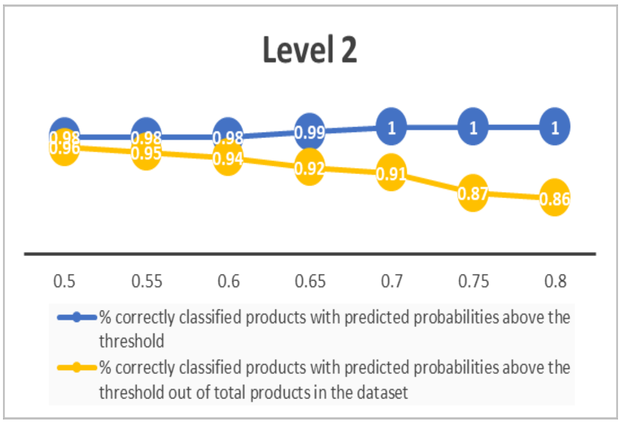


Fig. 2. Product Classification into three segments



This sensitivity report helps the business to decide the optimal threshold beyond which the error rate is ignorable enough that they would accept the predicted class of the product without any manual intervention. For instance, if the error rate of 99% is acceptable to business then they can finalize the threshold to be 0.65 for bakery products level 2.

Fig. 3. Level 2 Sensitivity Report

REFERENCES

[1] P. Cunningham, M. Cord and S. J. Delany, Supervised Learning.In Machine Learning Techniques for Multimedia Cognitive Technologies,2008, pp. 21-49

[2] T. Joachims. Text categorization with support vector machines: Learning with many relevant features. Technical Report 23, Universitat Dortmund, LS VIII,1997.

[3] Y. Yang. An evaluation of statistical approaches to text categorization. Technical Report CMU-CS-97-127, Carnegie Mellon University, April 1997