Hierarchical Classification

Mining Product Data from the Web

presented by

Bengi Koseoglu
Anjeza Gjuzi

submitted to the

Data and Web Science Group
Prof. Dr. Christian Bizer
University of Mannheim

July, 2018

Contents

Chapter 1

Introduction

E-commerce can be defined as 'the activity of buying or selling of products on online services or over the Internet'. Web, in a way changed our shopping behavior and became the root for emerging online business or shopping websites. Amazon was one of the first ecommerce web sites initially launched in 1995 for selling books and now has become the largest online retailer in the world offering more than 500 million products worldwide (Spacehero, 2018). By just looking at the number of products available on amazon, one can imagine the amount of products that is available in the web. This vast amount of information regarding products can be used to determine how product features influence product prices or how specific features of products influence the perception in customer reviews. The process of extracting knowledge from web using data mining techniques known as web mining, involves multiple steps; web crawling, product feature extraction, schema matching, multi-level classification, and sentiment analysis. This paper mainly focuses on our approach to hierarchical multi-level classification in our crawled product data, but it also involves giving an overview of e-shop selection, selection of seed products, construction of product catalog and web crawling as it is the basis of hierarchical classification.

Chapter 2

Preliminary Steps

Selecting Product Categories: First step of the project was selecting the product categories that are not similar to each other. As a group, we decided to focus on cameras for its deep structured product category and bags for its flat structure.

Deciding on E-Shops: Second step involved deciding on the which e-shops to crawl for. We decided to focus on e-commerce businesses that are not marketplace type, in English and located mainly in two countries, USA and the UK. In order to select e-shops for each main product category, we followed the approach proposed by Petrovski et al., and found the top 25 most frequently visited shopping web sites based on the ranking provided by Alexa for each product category (2016)

*Crawling and Feature Extraction*Third step was crawling and extracting information. We started the process by selecting 50 seed products for each category that are not too distinct from each other and then we crawled products according to the seed products. In total we have crawled and feature extracted 21 websites for cameras and 16 websites for bags.

Construct a product catalog: In the fourth step we constructed the product catalogs based on seed products. In order to build the product catalog, a web search using Google Shopping was done for each individual seed product. In the case of different data from different shops, the data provided by the web page of the original producer of the product, were chosen as the most correct ones. Each product was assigned a unique ID in the catalogs.

Chapter 3

Hierarchical Classification

3.1 Building Hierarchical Structure

First step for hierarchical classification was to built the hierarchy structure for both of our product categories; bags and cameras. According to Silla & Freitas, there are two different types of hierarchical structures; Directed Acyclic Graph (DAG) and tree structure (2010). The main difference is that in DAG, a node can have more than one parent node, whereas in the tree structure a node can have only one parent node. Having DAG structure limits the algorithms that we can use,therefore, we decided to apply tree structure. Below you may find the tree structure for cameras (Figure 3.1) and bags (Figure 3.2).

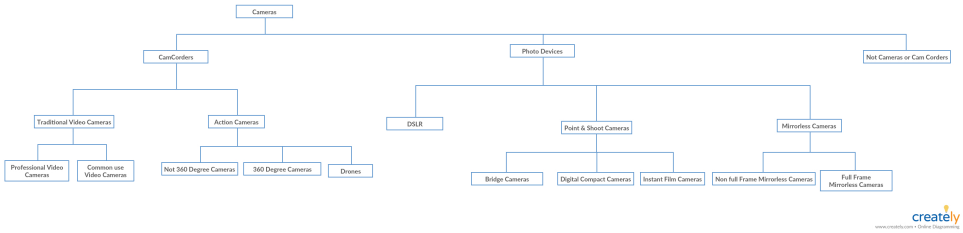

Figure 3.1: Tree Structure in Cameras

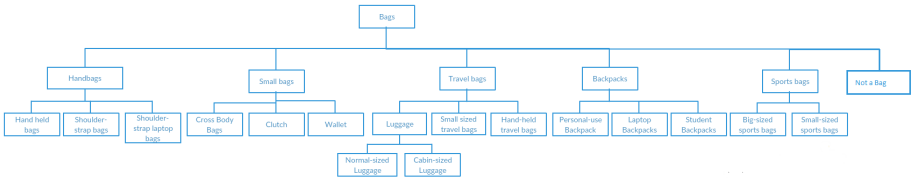

Figure 3.2: Tree Structure in Bags

3.2 Creating the Gold Standard

The second step was to create a gold standard in order to apply the algorithms aforementioned in step 3,2. Gold standard is a representation of our crawled data, therefore while annotating the products we included product types in the same proportions as the crawled ones. Our gold standard for cameras includes 418 manually classified camera product pages and for bags includes 610 manually classified bag product pages due to higher number of leaf nodes. Below you may see the distribution of the gold standards represented as pie charts.

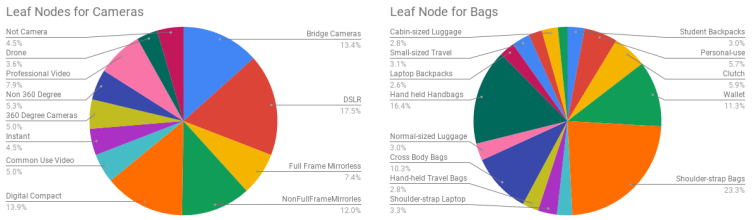

Figure 3.3: Leaf Node Distribution (Left: Cameras, Right: Bags)

3.3 Selecting Hierarchical Classification Methods

The third step was to decide was to decide which hierarchical classification methods to use. According to Silla & Freitas, there are three main approaches to hierarchical classification: flat classification, where one multi level classification model is applied on the leaf nodes, local classification, where multi level classification models are built through the structure in top-down manner and global classification, where a single multi-class and multi-label global model is built that takes the class hierarchy into account

(2010). Among these approaches, for the calculation of baseline, we decided to use flat classification and for the actual hierarchical classification we decided to use Local Classifier Per Node (LCPN) approach which is a type of local classification approach where a multi-class modeling is applied for each parent node in the class hierarchy in top-down manner, and global classification.

3.4 Setting the Baseline

The fourth step was creating a baseline for cameras and bags, in order to evaluate the applied algorithms against. For creating the baseline, we decided to apply flat classification approach. We applied decision tree algorithm with five fold cross validation only on the leaf nodes, while taking the name property of the entities as bags of words. As a result, the baseline for cameras defined as 0.6, whereas for bags defined as 0.72 accuracy score.

3.5 Classification

3.5.1 Local Classifier Per Parent Node

Only with Names as Bag of Words Initially our classification operated using only extracted names from schema.org annotations as bags of words and applying decision tree model with five key fold cross validation and parameter optimization. With this approach we reached 0.69 accuracy in cameras and 0.72 accuracy in bags in the leaf node level. By using the same variables, but with different algorithms, we improved our classification accuracy with 10 percent in cameras and 11 in bags.

Including Other Variables as Bag of Words Secondly, we used classification including brand, price and price currency to our initial model with only names. In case of missing values, we filled them with the most common value in that particular column. We applied different machine learning approaches using five key fold cross validation which included; decision trees, logistic regression, random forest and nearest centroid. In terms of decision tree algorithm, adding new variables to the model didn't improve the model in both bags and cameras. But changing the underlying algorithm show improved results. Among the algorithms that we have applied, both

in cameras and bags, we reached the highest accuracy in the leaf nodes with logistic regression algorithm with 0.82 and 0.84 accuracy in cameras and bags respectively. All the applied algorithms exceeded the baseline.

Logistic Regression with Feature Selection Thirdly, since logistic regression was the best algorithm to our particular problem at hand for both cameras and bags, we decided to focus more into improving this algorithm. As we look at the features that feed into our models, we realized that, once we convert our four extracted features (name, brand, price and price currency) into bags of words, our number of variables increases dramatically and not all of these variables might be useful to our prediction. Having redundant and irrelevant variables in the model, makes the model unnecessarily complex and may decrease the accuracy of the model (Brownlee,2016). In order to overcome this issue, we decided to apply feature selection before parameter optimization with grid search and modeling, and decided to chose the top 20 most important variables. For the feature selection algorithm, we decided to apply f-classification, which computes ANOVA F-value to a given sample. With this approach, we have reached 0.83 accuracy in cameras and 0.82 accuracy in bags. Compared to the previous logistic regression models, our accuracy increased only by 1 percent in cameras.

Logistic Regression with TF-IDF Fourthly, in our attempt to improve logistic regression, we decided to apply logistic regression with TF-IDF vectors, instead of BoW. A word has high TF-IDF in a document, if it appears a lot of times in the document but does not occur as much in the other documents (Calderon, 2018). With, TF-IDF we will be assigning a value to a particular variable between 0 and 1, which is different than BoW (Bag of Words), where values are assigned based on the frequency in the document. In a way, TF-IDF will act as feature selection by giving more weights to important and distinguishing variables. We applied logistic regression on TF-IDF weight assigned dataset with five k fold cross validation and grid search parameter optimization. As a result, with cameras and bags, we reached an accuracy of 0.77 and 0.82 respectively in node level, which is lower than logistic regression with BoW. This could be due to the fact that, even though a word occurs multiple times in the particular product page's extracted features and not occur as much in the other product pages' extracted features, this doesn't mean that it is relevant. Regardless of BoW's success over TF-IDF, we decided to pursue models with TF-IDF further, especially for the cases where a lot of variables exist.

Logistic Regression Adding HTML's and Feature Selection Lastly, in our attempt to improve logistic regression both for cameras and bags, we decided to take the whole HTML page that corresponds to the particular product page, preprocess and parse it, select features using f-classification and add it as tokens with TF-IDF weightings. We first found the corresponding HTML's and parsed it using beautiful soup package in python while removing HTML tags and annotations. After parsing, we preprocessed the data by removing noise, tokenizing, normalizing and eliminating tokens that have more than 45 characters since the longest word in English vocabulary has 45 characters. After preprocessing the dataset, we calculate TF-IDF weightings, apply f-classification feature selection in order to reduce number of variables to 100 and apply logistic regression with five k-fold cross validation and with grid search for parameter optimization. We applied this methodology for cameras at first to see whether it will work or not. As a result we reached an accuracy of 0.75 in leaf node level, which is still lower than logistic regression with extracted clean features and both with BoW and TF-IDF. We believe this is due to the fact that with adding .html pages, we are also adding noise to our model. Here, we also created another logistic regression model by adding an additional pre-processing step that compares the tokens after pre-processing against English dictionary. We thought this would improve our results since we are making sure that tokens are actual English words, unfortunately, this didn't help us improve the accuracy in leaf node level as the accuracy stayed almost the same as the previous model. Since both models resulted with comparably worse results, we decided not to pursue it further with bags.

3.5.2 Global Classification

So far we have applied flat classification approach for setting up our baseline and LCPN for actual classification. LCPN has some disadvantages such as en error at a certain class level is propagated downwards the hierarchy and complex structure of running multiple models(Silla & Freitas,2010) In order to overcome these issues, we decided to try global classification where only one multi label and multi class, classifier is built to discriminate all categories in a hierarchy simultaneously while takes class hierarchies into account. (Kiritchenko, Matwin, Nock, & Famili, 2006). For applying global classification, we followed the approach and steps proposed by Kiritchenko, Matwin, Nock, & Famili in their paper. These three steps include transforming the training data and making them consistent with a given class

hierarchy, application of regular learning algorithm on a multi-label dataset and re-labeling inconsistently classified test instances. For the multi level and multi class classification, we have only taken extracted names as BoWs and applied modified versions of support vector machine, Ada Boost and Logistic Regression classifiers that are capable of handling multi label multi class setting, using python's built in technique called 'one-vs-all'. This technique involves fitting one classifier per class and fitting the class against all the other classes for each classifier. After classification, there were some products where the algorithm assigned multiple leaf nodes or no leaf nodes. This is problematic, since according to our hierarchical tree structure one product must be belong to only one leaf node and cannot have multiple parents in same level. In order to fix this issue, we did a post-processing step by assigning the label with highest confidence score to the product. In cameras, with SVM,Ada Boost and Logistic Regression algorithms we reached an accuracy of 0.64,0.6 and 0.64, and in bags we reached an accuracy of 0.77, 0.72 and 0.73. Overall, global classification's outcome is better than our baseline, but worse than the LCPN.

3.6 Selecting the Best Model

In the appendix, you may find all the applied algorithms for cameras (table 2) and bags (table3). For cameras, LCPN approach with Logistic Regression, grid search and f classification for feature selection using using schema.org extracted features name, price, price currency and brand, resulted with the best accuracy of 0.83. For bags, the same combination without the feature selection is found the best model with 0.84 accuracy. These models yielded different accuracy in different levels of the hierarchy. At the top level, the best models in cameras and bags achieved an accuracy of 0.92 and 0.89. When we look at the top level in cameras, we see that 'not camera' category which involves products such as camera bags, solely sold camera lenses etc. is the hardest category to be distinguished. These products are hard to distinguish for the algorithm, because some lenses can have a price as high as cameras and most of the cameras are sold with the lenses, so the algorithm cannot distinguish these items from lens being 1 or 0 or from the price. Same phenomena can also be seen in the bag category of 'not a bag'. For instance, there are shoulder straps that are being sold for 1000 dollars, so the algorithm can not distinguish these items using price since they cost as much as some bags. After choosing the best algorithm, we also looked

some good positive and negative examples. In terms of cameras, non full frame and full frame cameras are distinguished well, drones are assigned into the action camera category but most of the time not to the drones leaf node and there are still problems distinguishing bridge cameras and digital compact cameras, but this isn't generally a shortage of the algorithm as most of the web sites also classified certain bridge cameras as digital compact cameras. In terms of bags, we observed that luggages are classified almost perfectly this might be due to having 'cabin' as a word in the name and laptop backpacks are still problematic since student backpacks usually have laptop area.

3.7 Applying the Best Model

We decided to apply above mentioned methodologies to the whole dataset. During application, we are first training our dataset with our gold standard and then applying the trained model to the whole dataset. So as a result, we made predictions for each parent and leaf node for the whole corpus. In order to assign the final predictions, we added one post processing steps, since we want the assigned labels to be consistent. This step involves taking the top level node prediction and assigning nodes that have the highest probability among the nodes that are the child nodes of the top level prediction. For instance, if a product's top level prediction is camcorders, for the second parent node level this product can either be classified as traditional video cameras or action cameras. Therefore, we looked at the assigned probability for both of these options and chose the one with the highest probability and assigned this label to the particular product. Below you may find the assigned labels presented as a pie chart.

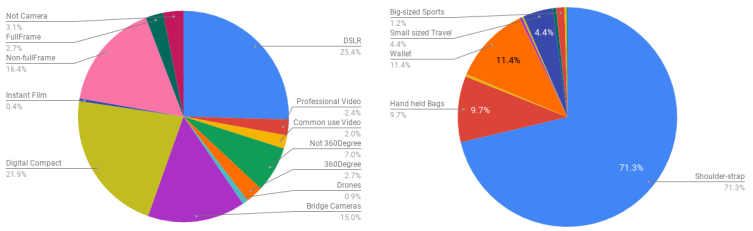

Figure 3.4: Leaf Node Distribution (Left: Cameras, Right: Bags)

Chapter 4

Conclusion

Web has in a way changed our shopping behavior and become the main source of information regarding products. This vast amount of information can be used to extract knowledge about products using web mining techniques which involves multiple steps; web crawling,product feature extraction, schema matching, multi-level classification, and sentiment analysis. This paper's goal was to give an overview of the previous steps which make the basis of our research and to describe our approach to multi-level classification. We started multi-level classification by first building hierarchical structure using tree structure, followed by selecting hierarchical classification methodologies which included flat classification, LCPN and global classification and creating the gold standard by manually annotating products. Then we continued the process by applying these selected approaches using different features with different machine learning algorithms. Among the tried approaches we got the best result in LCPN for both in bags and cameras, and among the tried feature combinations .HTML included models resulted with the worst outcomes since adding .HTMLs into the model added noise to the models. For cameras, LCPN approach coupled with logistic regression, grid search and f classification for feature selection using schema.org extracted features name, price, price currency and brand yielded with the best accuracy of 0.83. As for the cameras, LCPN approach, coupled with logistic regression and grid search using schema.org extracted features name, price, price currency and brand yielded with the best accuracy of 0.84. At the last step, we concluded our research by applying the best models to our whole corpus. The detailed results for each model can be found in the appendix and for more information you may check our jupyter notebooks.

Bibliography

[1] Petrovski, P., Primpeli, A., Meusel, R., and Bizer, C. (2016). The WDC Gold Standards for Product Feature Extraction and Product Matching. EC-Web.

[2] Silla, C. N., & Freitas, A. A. (2010). A survey of hierarchical classification across different application domains. Data Mining and Knowledge Discovery, 22(1-2), 31-72. doi:10.1007/s10618-010-0175-9

[3] Brownlee, J. (2016, October 30). An Introduction to Feature Selection. Retrieved July 7, 2018, from `https://machinelearningmastery.com/an-introduction-to-feature-selection`

[4] Kiritchenko, S., Matwin, S., Nock, R., & Famili, A. F. (2006). Learning and Evaluation in the Presence of Class Hierarchies: Application to Text Categorization. Advances in Artificial Intelligence Lecture Notes in Computer Science, 395-406. doi:10.1007/11766247_34

[5] Calderon, P. (2018, June 12). Bag of Words and Tf-idf Explained. Retrieved July 7, 2018, from `http://datameetsmedia.com/bag-of-words-tf-idf-explained/`

[6] ScrapeHero. (2018, February 01). How Many Products Does Amazon Sell Worldwide - January 2018. Retrieved July 14, 2018, from `https://www.scrapehero.com/how-many-products-amazon-sell-worldwide-january-2018/`

Appendix

Algorithms	Accuracy	Precision	Recall	F1
(Flat Classification) Only Names Decision Tree	0.6	0.62	0.60	0.6
(LCPN) Only Names Decision Tree w/ Grid Search	0.69	0.76	0.69	0.72
(LCPN) Names, Price, price currency, brand Decision Tree w/ Grid Search	0.68	0.77	0.68	0.71
(LCPN) Names, Price, price currency, brand Logistic Regression w/ Grid Search	0.82	0.88	0.82	0.84
(LCPN) Names, Price, price currency, brand Nearest Centroid	0.72	0.79	0.72	0.74
(LCPN) Names, Price, price currency, brand Random Forest	0.78	0.82	0.78	0.79
(LCPN) Names, Price, price currency, brand Logistic Regression w/ Grid Search & Feature Selection	0.83	0.88	0.83	0.85
(LCPN) Names, Price, price currency, brand Logistic Regression w/ Grid Search & TF-IDF	0.77	0.84	0.77	0.79
(LCPN) HTML Logistic Regression w/ Grid Search & TF-IDF & Feature Selection	0.75	0.85	0.75	0.77
(LCPN) HTML Logistic Regression w/ Grid Search & TF-IDF & Feature Selection & English Vocab	0.76	0.83	0.76	0.78
(Global Classification) SVM	0.64	0.63	0.64	0.61
(Global Classification) AdaBoost	0.6	0.61	0.6	0.58
(Global Classification) Logistic Regression	0.64	0.64	0.64	0.61

Table 1: Classification Results On Leaf Node Level for Cameras

Algorithms	Accuracy	Precision	Recall	F1
(Flat Classification) Only Names Decision Tree	0.72	0.76	0.72	0.72
(LCPN) Only Names Decision Tree w/ Grid Search	0.83	0.86	0.83	0.82
(LCPN) Names, Price, price currency, brand Decision Tree w/ Grid Search	0.82	0.86	0.82	0.82
(LCPN) Names, Price, price currency, brand Logistic Regression w/ Grid Search	0.84	0.85	0.84	0.84
(LCPN) Names, Price, price currency, brand Nearest Centroid	0.78	0.81	0.79	0.79
(LCPN) Names, Price, price currency, brand Random Forest	0.82	0.85	0.81	0.82
(LCPN) Names, Price, price currency, brand Logistic Regression w/ Grid Search & Feature Selection	0.82	0.84	0.82	0.82
(LCPN) Names, Price, price currency, brand Logistic Regression w/ Grid Search & TF-IDF	0.82	0.84	0.82	0.81
(Global Classification) SVM	0.77	0.8	0.77	0.77
(Global Classification) AdaBoost	0.72	0.77	0.72	0.71
(Global Classification) Logistic Regression	0.73	0.79	0.74	0.72

Table 2: Classification Results On Leaf Node Level for Bags