A System of Categorization and Classification Based on Certain Criteria

Robert Győrödi,
Department of Computer Science and Information Technology
University of Oradea,
Oradea, Romania
rgyorodi@uoradea.ro

Anamaria Tontea,
Computer Science,
University of Oradea,
Oradea, Romania
anamariatontea91@yahoo.com

Cornelia Győrödi,
Department of Computer Science and Information Technology
University of Oradea,
Oradea, Romania
cgyorodi@uoradea.ro

Livia Bandici
Faculty of Electrical Engineering and Information Technology
University of Oradea,
Oradea, Romania
lbandici@uoradea.ro

***Abstract*—This paper proposes a new system of categorization and classification using data mining techniques based on certain criteria/topics. We describe the design and implementation of proposed system that automatically categorizes a restaurant as being good or bad, using data mining techniques, based on users' reviews. For this study we took a data set consisting of approximately 9,000 reviews for 2,355 restaurants in Romania. The categorization was done on four criteria/topics: food, service, prices, atmosphere, and the results are represented as a range for each topic separately.**

***Index Terms* — Naïve Bayes, machine learning, NLTK framework, Mink, text categorization.**

I. Introduction

With many web sites appearing every day, the interest in the usage of machine learning on automatic text categorization is stimulated with this intensive growth of World Wide Web. Text categorization has the goal of deciding whether a document belongs to a set of predefined classes of documents [9] [10]. The problem of automatic document categorization is well known in information retrieval and usually tested on publicly available databases. A typical problem of information retrieval is to locate relevant documents based on user input data such as keywords or examples of documents [3].

The most important algorithms used to implement different solutions using machine learning systems are: decision trees, k-NN (k-Nearest Neighbor), Naïve Bayes, neural networks, SVM (Support Vector Machine) [4][5][6][11].

Usefulness of discovering new knowledge from existing information covers many different areas in society, and both functional organizations, such as companies and ordinary people.

Nowadays, the multitude and variation of existing restaurants raise us the following dilemma/question: "What's the best for me ?”. For this we proposed a system to automatically categorize a restaurant based on users' reviews. Its results can help tourists to find a suitable restaurant adequate to their own taste.

This classification is useful because it directly expresses the voice of the customer: they go to a restaurant, serve meal, and then can freely express their opinion through a comment about their thoughts regarding the restaurant. These comments, in time, come together and make it difficult to make a decision about choosing a restaurant, the main reason being that users do not have time to read all the reviews. Therefore, the users are being helped by these automatic classifications regarding the restaurants.

In this paper, we implemented a classification algorithm for categorization of restaurants. To build the model we chose the Naive Bayes classifier that was offered by the NLTK framework [1][6][7] and to build a database of reviews, we wrote a crawler to extract the information from a source (website http://www.gustos.ro/) that was selected due to the large number of reviews. In the next sections we will try our best to explain how we automatically categorize a restaurant as being good or bad, using data mining techniques, based on users' reviews.

II. Text data analysis and finding the relevant information

Most information retrieval systems support retrieval based on keywords and retrieval based on labels and on correlations (association). In the present paper retrievals were made based on n-grams, on 4 topics: food, service, atmosphere and prices, relative to two concepts: positive and negative

Systems based on keywords from documents collected sets of keywords or terms that frequently occur together and found relationships of association or correlation between them. Extraction was done through a preprocessing of the document, keyword detection (root words) and removal of linking words.

These are stored in a database similar to a transaction database on which associations mining algorithms are applied.

The choice of the four topics was made after a study based on reviews, and it was found that the people's main interest when choosing a restaurant is headed towards: the quality of the food, the way they are served, the prices and atmosphere. Also the study found that not all of them are interested in all 4 aspects, so that the results will be shown for each topic separately.

For belonging to one of the two concepts of the 4 topics, the reviews have been examined and from a visual analysis the most relevant n-grams giving it a negative or positive value, were decided. The next step was to build the vector of features and application of machine learning (ML) algorithm for categorization [4], that was described in more detail in the next section. In the categorization stage, the technologies we used are the TextBlob framework and NLTK framework from Python [1],[7].

III. Implementation

For this study we took a data set consisting of approximately 9,000 reviews for 2,355 restaurants in Romania. To build a database of reviews, we wrote a crawler to extract the information and we selected the http://www.gustos.ro/ website due to the large number of reviews. The categorization was done on four topics: food, service, prices, atmosphere, and the results were represented as a range for each topic separately. Expressions of a topic can oscillate between two concepts, one POSITIVE and one NEGATIVE. We say that an expression belongs to the positive concept, if it transmits user's satisfaction or gratitude. We say that an expression belongs to the negative concept if it shows the dissatisfaction of who wrote it. To build the model we have chosen the Naive Bayes classifier, provided by NLTK framework [1], [7]. Because we did not have training data, we had to work to get them. In Figure 1 we show the steps required for classifying reviews.

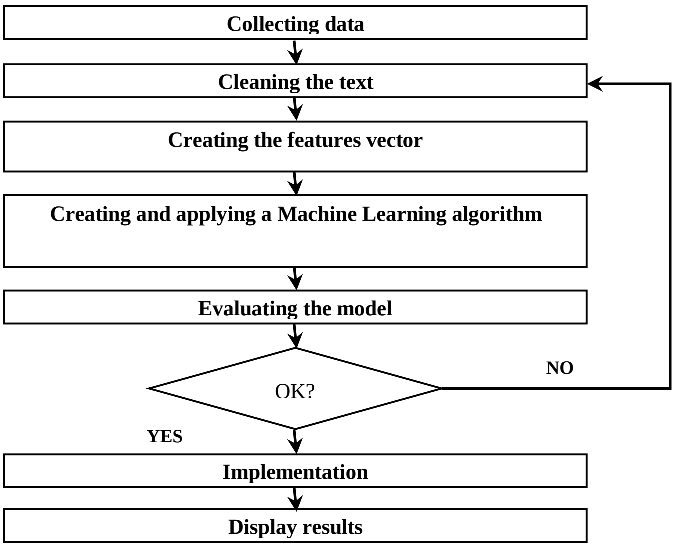


Fig. 1. Steps to achieve classification of reviews

Furthermore, we will detail the steps we considered necessary in the classification:

A. Collecting data

The first step was to get reviews. For this we wrote a crawler using PHP and Mink [2]. We have proposed the next steps to implement a crawler. These are described in the following pseudo code:

```
//get html source of this page
getPage ( 'http://www.gustos.ro/restaurante')
getcitiesLinks()
foreach cities
| get cityPage()
| get restaurantsLinks()
|
| foreach restaurantsLinks
| | get restaurantPage()
| | //get general information about restaurant:
name, city, address,
| | //phone, description
| | get generalInformation()
| | //MARKER_ID make the connection between review
and restaurant
| | get MARKER_ID
| | //insert restaurant in database
| | insert()
| | get reviewsZones()
| |
| | foreach reviewsZones
| | |MARKER_FIRMA =  MARKER_ID
| | |get review()
|//insert review in database
| | |insert()
| | |__
| |__
|__
```

B. Cleaning the text

This step is necessary for the formation of n-grams (unigrams, bigrams, trigrams; unigrams are expressions consisting of a single word, bigrams are expressions made up of two words and trigrams are expressions consisting of 3 words) and for removing everything that's irrelevant in processing the text.

This involves applying some steps, using the following techniques:

a) Tokenizing - helps to identify words and phrases by splitting them using delimiters such as space or other punctuation such as "!" "?" "." etc. In this step we implemented a program code by using the following pseudo code:

```
convert the review text to unicode
convert all html non breaking space(regex:
&\w+,foreg: &nbsp;) to space character
replace all type of white space(\r,\n,\t, etc.) to
single space character ' '
convert the review text to lower character
separating all words from text using delimiters:
space or other punctuation such as "!" "?" "."
```

```
eliminating words with small length(<=1) and which
didn't contain some character a-zA-Z
```

This step was implemented using Python, because of its facilities offered by its predefined functions. The review's text was divided into words by using some delimiters, with the help of *wordpunct_tokenize* from the NLTK library [3], but only after some processing using regular expressions is made:

```
from nltk.tokenize import wordpunct_tokenize
rev_words = set(wordpunct_tokenize(re.sub('\W+','
', re.sub('&\w+;','', processedReviews.lower()))))
```

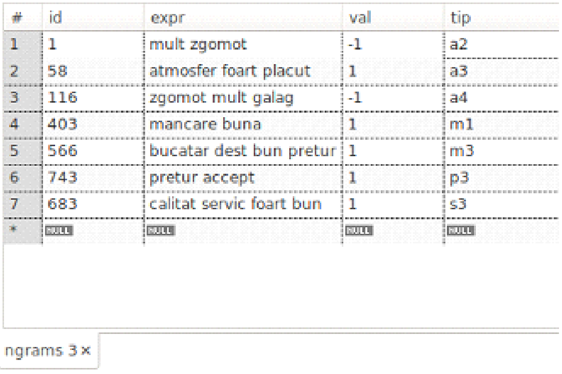
b) Elimination of irrelevant words (stopwords) - process in which irrelevant words are removed from the text in order to reduce redundancy of n-grams' expressions. In this way words that often appear in the text, but are still irrelevant, and even entangle the processing of the text, are being removed.

c) Stemming - At this stage the words are replaced by their roots, by removing their suffixes and prefixes

C. *Finding characteristics*

This step is about constructing, based on the previously processed reviews, of unigrams, bigrams, trigrams and 4grams. The results were inserted into a MySQL table, and besides the expression's name, the frequency of appearance was also calculated.

Then the most common 100-250 expressions were selected from the table for each topic separately, also for bigrams, for trigrams and 4grams. From this selection, there have been inspected, analyzed and selected only the phrases relevant to the topic at hand, the irrelevant ones being deleted. Then they were assigned to one of the 2 classes/concepts: POSITIVE and NEGATIVE, being represented by "1" for a positive expression and "-1" for negative expression. Also in another column of the table has been saved, for each expression, their belonging to one of the four topics, more precisely the expression type as shown in Figure 2:



#	id	expr	val	tip
1	1	mult zgomot	-1	a2
2	58	atmosfer foart placut	1	a3
3	116	zgomot mult galag	-1	a4
4	403	mancare buna	1	m1
5	566	bucatar dest bun pretur	1	m3
6	743	pretur accept	1	p3
7	683	calitat servic foart bun	1	s3
*	NULL	NULL	NULL	NULL

Fig. 2. The structure of the n-grams table

a2 = bigrams for environment topic

a3 = trigrams for environment topic

a4 = expressions consisting of 4 words for the environment topic

m1 = bigrams for food topic

m3 = trigrams for food topic

p3 = trigrams for prices topic

s3 = trigrams for serving topic

D. *Classifying uncategorized reviews using semi-automatic classification*

The reviews categorized according to the above algorithm was considered the input data of the training set to categorize the reviews remaining unclassified. So the model training was done by calling the NaiveBayesClassifier() method, and the model accuracy was calculated with the accuracy() method. For this, we divided all the data in: 80% training data and 20% testing data. We have implemented the following code to calculate accuracy, and to display the most relevant terms and the weights associated with each class.

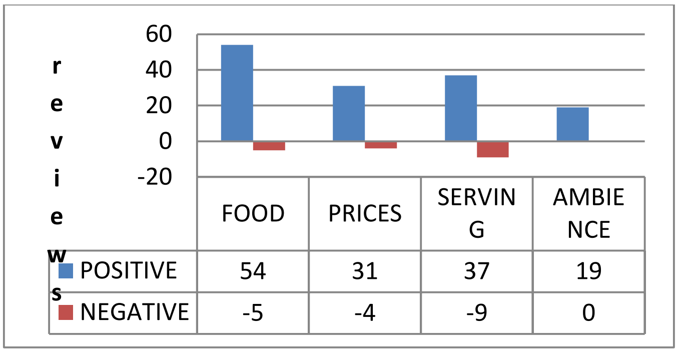
```
#Preturi
reviewsPriceTrain,reviewsPriceTest=fromMysql.
     TakeFromDBReviewsLIST_to_Train('preturi')
rev_to_classify=fromMysql.TakeFromDBReviewsLIST_to_Classi
fy('preturi','preturi')
clTextBlobPreturi =
     NaiveBayesClassifier(reviewsPriceTrain)
print("Accuracy: {0}".
     format(clTextBlobPreturi.accuracy(
        reviewsPriceTest)))

printclTextBlobPreturi.show_informative_features(200)
```

Then, with the *clTextBlobPreturi.classify(rev)* function we classified one review.

```
for key,rev in rev_to_classify:
predicted=clTextBlobPreturi.classify(rev)
fromMysql.UpdatePredictedValues(key,predicted,"Pri
ce_Predicted_ALL_TB")
print "Predicted: "+str(predicted)+"; rev: "+rev
```

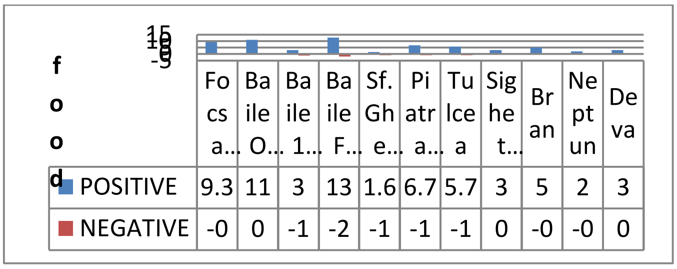
We have obtained 0.920577617329 for accuracy of the price topic. So after this step most reviews were classified and for each company the average on each topic separately was calculated. The results were stored in a new table where we could view the final classification for each restaurant.

The results are shown in Figure 3, Figure 4, Figure 5, Figure 6 and Figure 7. Each figure shows the final classification for each restaurant from the four points of view: the food, the ambience, the serving and prices.



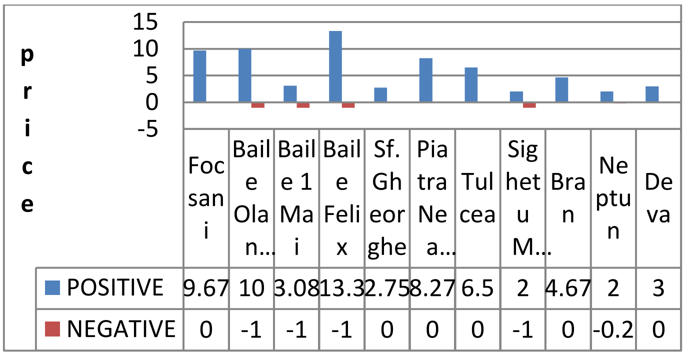
	FOOD	PRICES	SERVING	AMBIENCE
🟦 POSITIVE	54	31	37	19
🟥 NEGATIVE	-5	-4	-9	0

Fig. 3. Classification by reviews



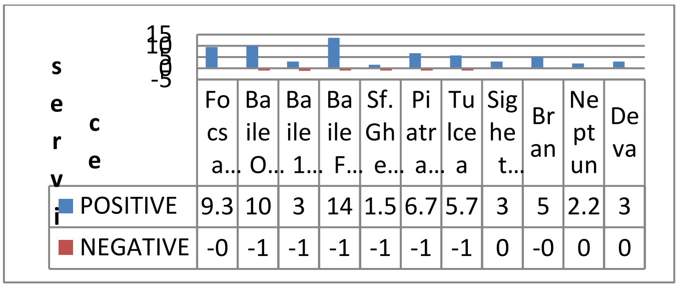
	Focsa...	Baile O...	Baile 1...	Baile F...	Sf. Ghe...	Piatra...	Tulcea	Sighet...	Bran	Neptun	Deva
🟦 POSITIVE	9.3	11	3	13	1.6	6.7	5.7	3	5	2	3
🟥 NEGATIVE	-0	0	-1	-2	-1	-1	-1	0	-0	-0	0

Fig. 4. Classification by food topic



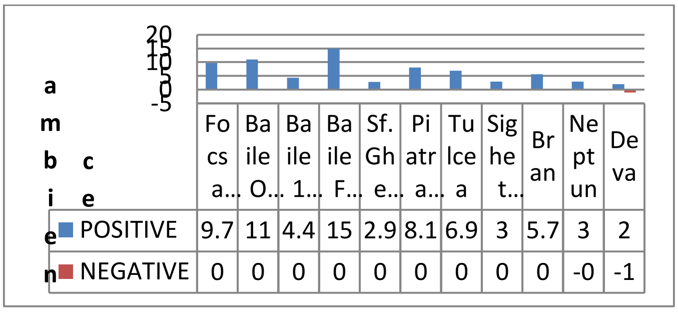
	Focsani	Baile Olan...	Baile 1 Mai	Baile Felix	Sf. Gheorghe	Piatra Ne a...	Tulcea	Sighetu M...	Bran	Neptun	Deva
🟦 POSITIVE	9.67	10	3.08	13.3	2.75	8.27	6.5	2	4.67	2	3
🟥 NEGATIVE	0	-1	-1	-1	0	0	0	-1	0	-0.2	0

Fig. 5. Classification by price topic



	Focsa...	Baile O...	Baile 1...	Baile F...	Sf. Ghe...	Piatra...	Tulcea	Sighet...	Bran	Neptun	Deva
🟦 POSITIVE	9.3	10	3	14	1.5	6.7	5.7	3	5	2.2	3
🟥 NEGATIVE	-0	-1	-1	-1	-1	-1	-1	0	-0	0	0

Fig. 6. Classification by service topic



	Focsa...	Baile O...	Baile 1...	Baile F...	Sf. Ghe...	Piatra...	Tulcea	Sighet...	Bran	Neptun	Deva
🟦 POSITIVE	9.7	11	4.4	15	2.9	8.1	6.9	3	5.7	3	2
🟥 NEGATIVE	0	0	0	0	0	0	0	0	0	-0	-1

Fig. 7. Classification by ambience topic

IV. Conclusions

The aim of this work was to create a system to automatically categorize a restaurant based on users' reviews present in the database and possible future reviews, using data mining techniques. The categorization was done on four criteria/topics: food, service, prices, atmosphere, and the results were represented as a range for each topic separately.

The advantage of this method proposed in the paper is that the threads are independent of each other and could be used as training sets for future reviews. The training sets of the 4 threads could be reused successfully in categorizing new restaurants on other websites. Another advantage would be the reuse of the categorization model for other areas.

The crawler implemented in this paper has better efficiency due to the functions of the Mink framework [2] that contains special methods for crawler operation. The tests carried out showed that the proposed method has good accuracy. The results of this system could help tourists to find a suitable restaurant adequate to their own taste, or could be of great help in the restaurant's marketing.

The method proposed in this paper could be extended in the future to the collection of characteristics for topics, as bringing more reviews in the process entails expanding list of features.

CONFLICT OF INTERESTS

The authors declare that there is no conflict of interests regarding the publication of this paper.

ACKNOWLEDGMENT

This work was performed through the Partnerships Program in priority areas, PN-II-PT-PCCA-2013-4-2225 - No. 170/2014 developed with the support of MEN - UEFISCDI, "Electromagnetic methods to improve processes wine".

References

[1] L. Steven, "Simple Text Classification with Python and TextBlob", Aug 26, 2013, on-line: http://stevenloria.com/how-to-build-a-text-classification-system-with-python-and-textblob/

[2] K. Kudryashov, "Mink documentation, Release 1.6", Nov 03 2015, on-line https://media.readthedocs.org/pdf/mink/latest/mink.pdf

[3] H. Zhang, "The Optimality of Naive Bayes". Proceedings of the 17th International FLAIRS Conference 2004, Menlo Park, CA., AAAI Press, 2004.

[4] R. Győrödi, I. Lungu I., C. Győrödi, "Sisteme avansate de descoperire a cunoștințelor din bazele de date", Editura Universității din Oradea, 2012, ISBN 978-606-10-0733-2.

[5] S. Russell, P. Norvig, (2003). "Artificial Intelligence: A Modern Approach", (2nd ed.). Prentice Hall. ISBN 978-0137903955, 2003.

[6] G. I. Webb, J. Boughton, Z. Wang, (2005). "Not So Naive Bayes: Aggregating One-Dependence Estimators. Machine Learning", Ed. Springer, 58 (1): 5–24. doi:10.1007/s10994-005-4258-6.

[7] M. Lutz, "Learning Python, 5th Edition Powerful Object-Oriented Programming", Published by O'Reilly Media, Inc., 1005 Gravenstein Highway North, Sebastopol, CA 95472, 2013.

[8] E. Seyda, C. Lee Giles, "A Comparative Study on Representation of Web Pages in Automatic Text Categorization", IIS-IST Technical Report-62003, The Pennsylvania State University, University Park, June 2003,

on-line: http://web.mit.edu/seyda/www/Papers/IIS-IST-TR_webmining.pdf.

[9] E. Glover, K. Tsioutsiouliklis, S. Lawrence, D. M. Pennock, G. W. Flake, “Using Web Structure for Classifying and Describing Web Pages”, The Eleventh International World Wide Web Conference, May 7-11, 2002, Hawaii, USA, ACM 1-58113-449-5/02/0005.

[10] E. Glover, G. Flake, S. Lawrence, W. P Birmingham, Kruger, A., C. L Giles and D. Pennock, “Improving category specific web search by learning query modifications”. Symposium on Applications and the Internet, SAINT 2001, San Diego, California, January 8–12, IEEE Computer Society, Los Alamitos, CA, pp. 23–31, 2001.

[11] C. Győrödi, R. Győrödi, G. Pecherle, G. M. Cornea – “ Full-Text Search Engine Using MySQL”, International Journal of Computers, Communications & Control (IJCCC), Vol. 5, Issue 5, December 2010, ISSN 1841-9836, pag. 731-740.