HARAMAYA UNIVERSITY

SCHOOL OF GRADUATE STUDIES

COLLAGE OF COMPUTING AND INFORMATICS

DEPARTMENT OF COMPUTER SCIENCE

NATURAL LANGUAGE PROCESSING

Project Report

Designing and Developing Comparative Document Classifier

"The case of SVM, KNN, Gaussian Naive Bayes & Perceptron Algorithms"

By

Suleiman M. A. Gargaare

&

Abdirkadir H. Aden

TABLE OF CONTENTS

INTRODUCTION

Document classification has been a vital research area or topic since the establishment of digital documents have been widely spready [1]. Nowadays, text classification is an important task because of the very large amount of text documents that we need to deal daily. In general, document classification can be classified topic-based document classification and document genre-based classification. Topic-based document categorization can be classified documents according to their topics [2]. Also, texts can be written in many different genres, for example, academic papers, advertisement updates, political news and movie reviews. Genre is referred on the way a text was made, the way it was modified, the identification of language it uses, and the type of listeners to whom it is addressed. Existing study on genre classification have found that this task differs from categorization of topic-based [3]. Commonly, most data based on genre classification are collected from the newspaper, web, noticeboards, and live broadcast.

The classification can be information retrieval from the metadata, manually classifying, or via an automatic classifier retrieving information from the content. As manually classifying documents can be a time-consuming and in-consistent task, it is usually not beneficial on a larger scale. Instead, Automatic Document Classification is suggested to solve this kind of task, as it is an automatic process that can be used on for larger systems [4]. The document classification contains many concepts as *Figure 1.1.* shows.

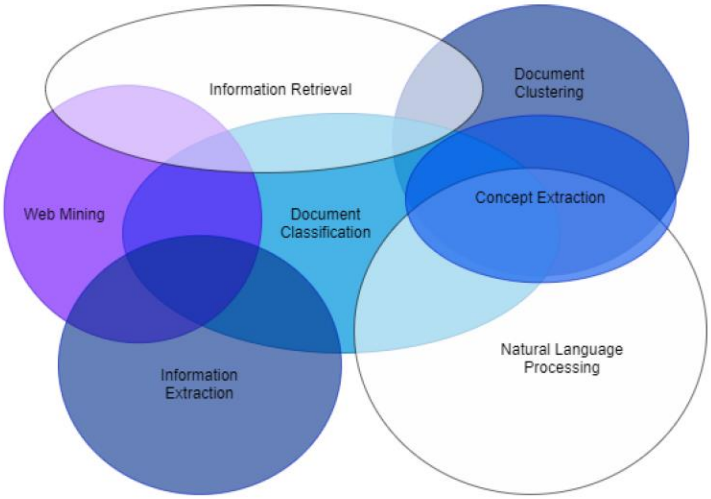


Figure 1.1. By [5], *Venn diagram of the Text Mining area.*

This report is organized as follows, In Section 1 Introduction has been presented. Section 2 presents different types of Classification Algorithms. Section 3 describes the related researches of the project. Section 4 introduces the Methodology. Section 5 shows Evaluations and Results of the experimental project. Conclusions is discussed in section 6 and Reference is cited in Section 7.

CLASSIFICATION ALGORITHMS

Logistic Regression

The binary outcomes such as either something happens, or doesn’t, yes or not, pass or fail, alive or dead calculated by logistic regression according to [6]. Also, it described two variables; Independent and dependent variables that are analyzed to decide the binary outcome through the outcomes falling into either numeric or categorical. The independent variables be able to be categorical or numeric, but the dependent variable is permanently categorical and it’s written like this:

$$P(A=1|B) \text{ or } P(A=0|B)$$

Whereas, Y and X calculates probability of dependent variable and independent variable consequently.

Positive or negative connotation {0,1} word or tree, grass and flower which is a common object contained in a photo calculated by this probability to shown probability of each object between 1 and 0 [6].

K. Nearest Neighbor

In the KNN objects are classified as [7] defines filled by selection of several labeled training examples with their smallest distance from each object. The k-nearest neighbor classification method is outstanding with its simplicity and is typically used techniques for text classification. Even though it requires more time for classifying objects when a large number of training examples are given, this method accomplishes well even in controlling the classification tasks with multi-categorized documents. KNN should hand-picked some of them by figuring the distance of each test objects with all of the training examples.

Decision Trees

According to [8] a Decision Tree is a tree in which internal nodes are labeled by terms. While weight of term has or numerical data labeled the branch and categories labeled the leaf. In the Decision Tree concepts using ‘divide and conquer’ strategy. Also, [8] defines that each node in a tree is associated with set of cases. According to this whole training examples should have to checked weather under the same label or not. Then selects partitioning term from the pooled classes of documents that have same values for term and place each of such class in a separate subtree, if not same label.

Random Forest

As [6] stated, the random forest algorithm is an expansion of decision tree. in that, in a real world constructing some axis decision tree within training data. It basically, norms your data to bond it to the nearby tree on the data scale. Random forest prototypes are useful as they cure for the decision tree’s problem of “forcing” data points within a category unnecessarily.

Naive Bayes Algorithm

Naive Bayes classifier in [9] is a “simple probabilistic classifier based on applying Bayes’ Theorem with strong independence assumptions”. It computes the subsequent probability of the document goes to different classes and assigns document to the class with the maximum subsequent probability. This probability model would be autonomous feature model. Thus, existing of one feature does not affect other features in classification tasks.

Perceptron

A perceptron is a neuron that is artificial in which the threshold function is an activation function. Think about an artificial neuron having z_1, z_2, …, z_n as the input signals and m_1, m_2, …, m_n as the associated weights. Let we consider m_0 for constant [10]. If the output of neuron is given by the bellow function let, we consider as a perceptron.

$$o(x_1, x_2, \ldots, x_n) = \begin{cases} 1 & \text{if } w_0 + w_1 x_1 + \cdots + w_n x_n > 0 \\ -1 & \text{if } w_0 + w_1 x_1 + \cdots + w_n x_n \leq 0 \end{cases}$$

Support Vector Machines

For text classification task Support Vector Machine that has been widely and fruitfully used. SVM is supervised classification algorithm. High dimensional input space: In text classifiers, one has to deal through huge number of features [11]. SVM has potential to handle large feature spaces, Since SVM use over appropriate protection that does not necessarily depend on the number of features, in addition, the most of text categorization problems are linearly separable: for this reason, the idea of SVMs is to find and search such kind of linear separators.

Gaussian Naive Bayes

As [12] defined, GaussianNB is based on Gaussian Normal Distribution and supports continuous data. It is a variant of Naive Bayes. Often continuous values associated with each class and distributed according to normal distribution to compute continuous data.

RELATED RESEARCHES

Nowadays, KNN is the most researched topic by most of the researchers where the major aspect is to perform a detailed study over survey applications that are performed by implementing introductory data mining books and the reports of surveys that are performed as specified in the survey article documented by [13] which proposed many improvements of KNN algorithms for implementing classification of data. Another interesting publication in [14] was performed on the weighted KNN classification algorithm that is based on various symbolic features, in which the distance is measured and calculated then depicted in the form of tables to produce real-valued distances from symbolic domains that also represents features. The authors propose that the proposed algorithm is superior when compared with the existing algorithm like KNN as it is implemented on three distinct application domains whose major advance label is the possession of training speed and simplicity in implementation.

[15] publishes his study that focuses on an adjustment of weight while implementing KNN for identifying optimum weighted vector by using an optimization function that is based on the “leave-out-out cross-validation” technique and “greedy hill climbing technique” and on the same hand the work introduced three major “decision tree algorithms” [16] and many other studies provide comprehensive surveys being performed on the applications that are based on distinct decision tree algorithms in the fields of machine learning and data mining approaches.

SVMs are another classification approach which functions by analyzing a feature space and attempting to construct a hyper plane to separate data points belonging to different classes [17]. They operate by mapping data onto a higher dimensional space using a kernel function and defining the hyper plane there. Although SVMs are inherently binary classifiers they can be modified for multiclass problems by using pair wise classification, which tackles a problem as a series of binary problems.

SVMs and their mathematical background are illustrated in [17] where the survey is implemented in the comprehensive book [15] that provides illustration in support vector machines that comprises of self-learning kernels and the other publication that has cited major aspects of SVMs are applied over implementation of text categorization in [18].

Decision Trees are another approach which many researchers employ in tackling the problem of automatic affect recognition. This is a simplistic classifier which makes observations on data and maps these observations to decisions on class ownership [17]. It functions by constantly querying a test instance to gain more information about which class it may belong through a combination of if-then rules.

METHODOLOGY

Experimental Setup

To do this work we have chosen four different machine learning classification algorithms and they are, K. Nearest Neighbor (KNN), Support Vector Machine (SVM), Perceptron and Gaussian Naïve Bayes.

In this experimental project we have used these classification algorithms to analyze and evaluate their efficiency and effectiveness. We have implemented these algorithms using Python with different libraries and modules including; Pandas, Scikit-learn, NumPy, and Matplotlib

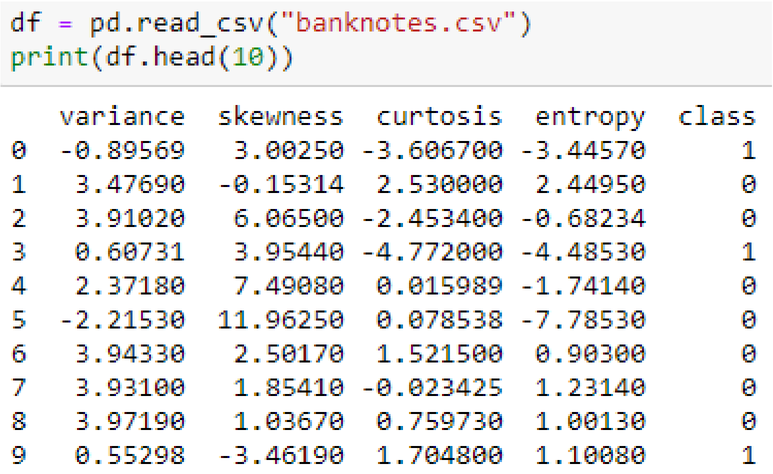
So, for the evaluations of these algorithms we have used the different types of Confusion Metrics and in details it is explained below.

In addition, we have used excel spreadsheet to present the results of evaluation and analysis that we got during the experimental work. In the analysis part we will present the findings and results of KNN, SVM, Perceptron and GaussianNB algorithms in terms of their accuracy, precision, recall, error rate and f-score by illustrating figures or charts.

Data Collection and Analysis

This experimental project has been used on a whole bunch of data set called Bank Notes provided by UC Irvine [19] which has information about different banknotes. The UC Irvine has built this dataset by took pictures of various different banknotes and measured various different properties of that banknotes and in particular, they categorized each of these banknotes as either counterfeit banknotes or not counterfeit (authentic). We have used this dataset to get the exact performance of every algorithm.

The dataset consists of one thousand three hundred and seventy-three rows and five columns that employed to perform 50% as a training set and 50% as a testing set of the model. *Figure 1.3* illustrates the first 10 rows in banknotes dataset that we have used to train and test the model.



```
df = pd.read_csv("banknotes.csv")
print(df.head(10))
```

```
   variance  skewness  curtosis  entropy  class
0  -0.89569   3.00250 -3.606700 -3.44570      1
1   3.47690  -0.15314  2.530000  2.44950      0
2   3.91020   6.06500 -2.453400 -0.68234      0
3   0.60731   3.95440 -4.772000 -4.48530      1
4   2.37180   7.49080  0.015989 -1.74140      0
5  -2.21530  11.96250  0.078538 -7.78530      0
6   3.94330   2.50170  1.521500  0.90300      0
7   3.93100   1.85410 -0.023425  1.23140      0
8   3.97190   1.03670  0.759730  1.00130      0
9   0.55298  -3.46190  1.704800  1.10080      1
```

Figure 1.3. First 10 rows of banknotes.csv

Each row of this dataset represents on banknote and has four different input values and these inputs has an output value 0 or 1. 0 meaning it was a genuine (authentic) bill and 1 meaning it was a counterfeit bill.

So, this experiment has been used supervised learning to begin to predict or model some sort of function that can take four values as input and predict what the output would be.

The model has been built using Python language, Jupyter Notebook as compiler and different libraries including; Pandas, Scikit-learn, and Matplotlib.

EVALUATIONS AND RESULTS

In the evaluations, we have divided the data set into two different sets, training and testing datasets. We have used training data set so as to build the model (classifier) and then we have tested this classifier to do the prediction. As we mentioned in the experiment section, this splitting is normally 50% training dataset and 50% test dataset.

Confusion Matrix

Confusion Matrix is also known as an Error Matrix, it is a table that is used to define the analyses of a classification algorithms on a set of test data for which the true values are well known. Actually, it is a table that has two dimensions; Actual Value and predicted Value as Table 1.1 illustrates.

Actual	**Predicted**		
		No	**Yes**
	No	*TN*	*FP*
	Yes	*FN*	*TP*

Table 1.1: Truth Table of Confusion-Matrix

TN: True-Negative, *TP*: True-Positive, *FN*: False-Negative, *FP*: False-Positive.

So, the confusion matrixes of this experimental project's classification algorithms will be as illustrated the below matrixes.

$$\begin{matrix} 363 & 1 \\ 0 & 322 \end{matrix}$$

Confusion Matrix 1.1 KNN

$$\begin{matrix} 389 & 7 \\ 0 & 290 \end{matrix}$$

Confusion Matrix 1.2 SVM

$$\begin{matrix} 382 & 1 \\ 9 & 294 \end{matrix}$$

Confusion Matrix 1.3 Perceptron

$$\begin{matrix} 348 & 39 \\ 60 & 239 \end{matrix}$$

Confusion Matrix 1.4 GaussianNB

The evaluation metrics can be classified in Accuracy, Precision, Recall, Error Rate and F-Score.

Accuracy

It is how close the measured value to the actual (true) value.

$$\text{Accuracy} = \left(\frac{\text{TP+TN}}{\text{Total Tuples in Test Dataset}}\right)$$

The experiment has got the accuracy of these four algorithms; KNN, SVM, Perceptron and GaussianNB have scored 99.85%, 98.98%, 98.54% and 85.57% respectively as *figure 1.4* shows.

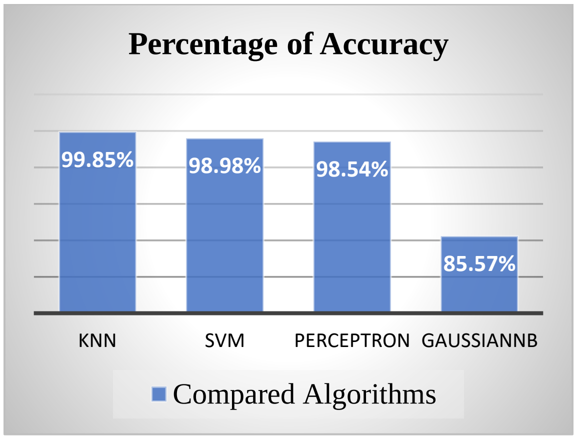


Figure 1.4. Percentage of Accuracy

Precision

It is an evaluation analysis technique that finds how close the measured values are to each other.

$$\text{Precision} = \left(\frac{\text{TP}}{\text{Predicted Yes}}\right)$$

In the precisions these algorithms; KNN, SVM, Perceptron and GaussianNB have got 0.996, 0.976, 0.996 and 0.856 respectively as *figure 1.5* illustrates.

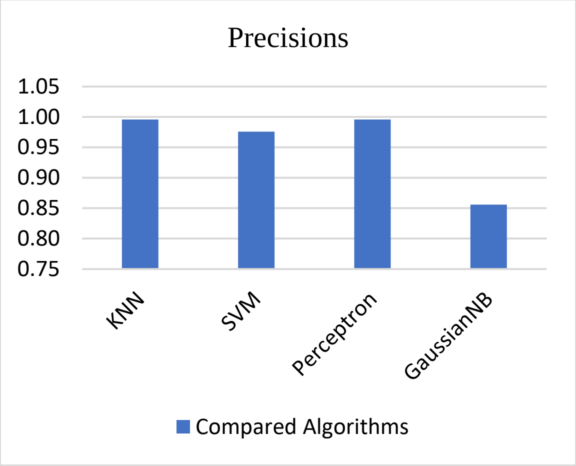


Figure 1.5. Precisions of Compared Algorithms

Recall

It is the ratio of all correctly predicted positive predictions.

$$\text{Recall} = \left(\frac{\text{TP}}{\text{Actual Yes}}\right)$$

In the recall evaluation these algorithms; KNN, SVM, Perceptron and GaussianNB have reached 100%, 100%, 97% and 79% respectively as *figure 1.6* notes.

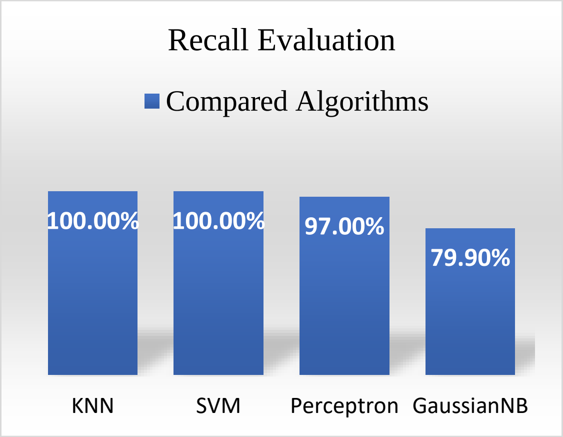


Figure 1.6. Recall of Compared Algorithms

Error Rate

It is an evaluation analysis technique that calculates the number of all incorrect predictions divided by the total number of the datasets.

The worst error rate is 1.0 and the best error rate is 0.0

$$\text{Error Rate} = 1 - \text{Accuracy}$$

In the Error Rate these algorithms; KNN, SVM, Perceptron and GaussianNB have made 0.002, 0.011, 0.015 and 0.145 respectively as *figure 1.7* declares.

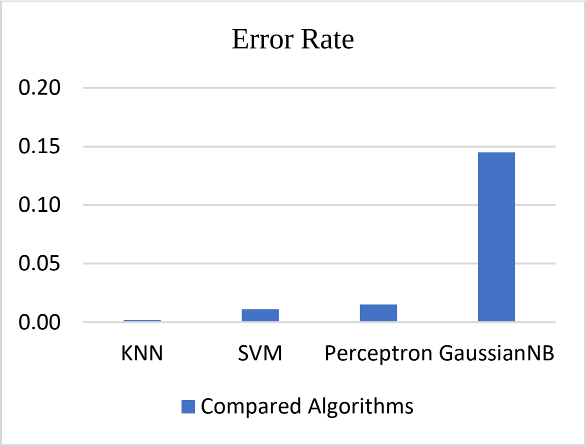


Figure 1.7. Error Rate of Compared Algorithms

F-Score

It is evaluation analysis technique that calculates the harmonic mean of precision and recall.

$$\text{F-Score} = \left(\frac{2(P*R)}{P+R}\right)$$ Where *P* is a Precision and *R* is a Recall.

In the F-Score evaluation these algorithms; KNN, SVM, Perceptron and GaussianNB have marked 0.997, 0.987, 0.982 and 0.826 respectively as *figure 1.8.* states.

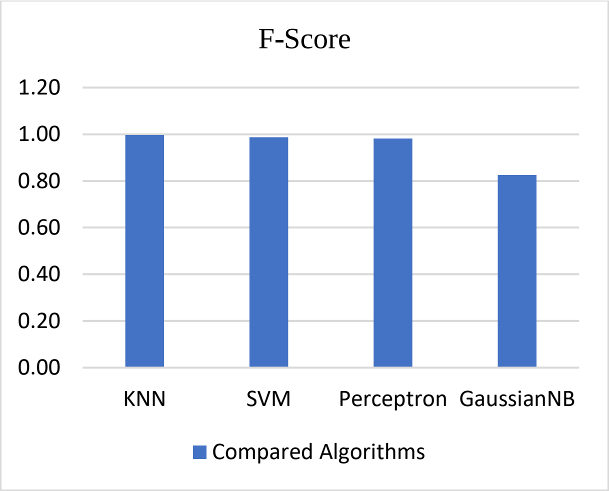


Figure 1.8. F-Score of Compared Algorithms

CONCLUSION

In this comparative experiment, we have compared the analyses and performance of various classification algorithms or classifiers; KNN, Perceptron, SVM and Gaussian. Bank Notes data set created by UC Irvine has been used as experiment. Numbers of cross-folds in each case were 10. In addition, this project has focused on identifying the better algorithm for document classification that works well on different data sets. However, it has found that the accuracies of the tools depending on the data set used. Also, it has been noted that the classifiers of a special group also did not perform with equal accuracies.

In terms of overall performance that is if we consider the accuracy of KNN, SVM, Perceptron and Gaussian. The SVM and KNN Algorithms performs better than the others in this experiment. We also compared these classifiers with the help of Confusion Matrix.

When we have reviewed on different types of algorithms and compared them, it can be concluded that SVM and KNN classifiers has been recognized as two better algorithms for document classification.

Furthermore, the analyses result of this experiment identified that the performance of the algorithms depends on the data set that you have, especially on the number of features used in the data set. So, we would like to recommend, researchers, academicians and students to try their data set on a different of algorithms and try to choose the best on.

To improve the performance and accuracy of the document classification algorithms, future study is required.

REFERENCES

[1] M. Ikonomakis, S. Kotsiantis, V. Tampakas, "Text Classification Using Machine Learning Techniques," *WSEAS TRANSACTIONS on COMPUTERS,* vol. 4, no. 8, pp. 966-974, 2005.

[2] Y. Yang, "An evaluation of statistical approaches to text categorization," *Journal of Information Retrieval,* vol. 1, p. 67–88, 1999.

[3] B. Kessler, G. Nunberg, and H. Schutze, "Automatic detection of text genre," *In Proceedings of the Thirty-Fifth ACL and EACL,* p. 32–38, 1997.

[4] Goller, Löning, Will, Wolff, "Automatic Document Classification.," *Internationalen Symposiums für Informationswissenschaft,* p. 145 – 162, 2000.

[5] M. T. U. U. Hugo Moritz, A Comparative Study of Machine Learning Algorithms for Document Classification, Uppsala: Unpublished, 2020.

[6] R. Wolff, "Classification Algorithms in Machine Learning," https://monkeylearn.com/, Last Accessed, Feb 2021, 2020.

[7] Tam, Santoso A, Setiono R, "A comparative study of centroid-based, neighborhood-based and statistical approaches for effective document categorization," in *ICPR '02 Proceedings of the 16 th International Conference on Pattern Recognition (ICPR'02) ,vol.4 , no. 4 , 2002, pp.235–238.*, 2002.

[8] Russell Greiner, Jonathan Schaffer, "Exploratorium – Decision Trees," in *http://www.cs.ualberta.ca/~aixplore/learning/ Decision Trees. Last Accessed, Jan 2021*, Canada, 2001.

[9] I. Rish, "An Empirical Study of the Naïve Bayes Classifier," *Proc. of the IJCAI-01 Workshop on Empirical Methods in Artificial Intelligence.,* pp. citeulike-article-id:352583, 2001.

[10] V. N. Krishnachandran, Machine Learning, Kerala, India: Vidya Centre for Artificial Intelligence Research, 2018.

[11] Pratiksha P. Pawar, S. H. Gawande, "A Comparative Study on Different Types of Approaches to Text Categorization," *International Journal of Machine Learning and Computing,* vol. 2, pp. 423 - 426, 2012.

[12] P. Majumder, "Gaussian Naive Bayes," https://iq.opengenus.org/gaussian-naive-bayes/. Last Accessed, Dec 2020, India, 2018.

[13] Agrawal, R., T. Imielinski and A.N. Swami, "Database Mining: A Performance Perspective," *IEEE Trans. Knowledge and Data Engineering,* vol. 6, pp. 914-925, 1993.

[14] Jiawei Han, Jian Pei, Micheline Kamber, Data Mining: Concepts and Techniques, 2nd Edition: Morgan Kaufmann, 2006.

[15] H. Z., A Short Introduction to Data Mining and Its Applications, IEEE, 2011.

[16] B. Rama, P. Praveen, H. Sinha and T. Choudhury, "A study on causal rule discovery with PC algorithm," in *International Conference on Infocom Technologies and Unmanned Systems (Trends and Future Directions) (ICTUS),*, Dubai, 2017.

[17] S.Nagaparameshwara Chary, B Rama, "A Research Travelogue on Classification Algorithms using R Programming," *International Journal of Recent Technology and Engineering (IJRTE),* vol. 8, no. 4, pp. 9155 - 9158, 2019.

[18] P. Praveen, B. Rama, Uma N. Dulhare,, "A study on monothetic Divisive Hierarchical Clustering Method," *International Journal of Advanced Scientific Technologies Engineering and Management Sciences,* vol. 3, pp. ISSN 2454-356X, 2017.

[19] L. Volker, Bank Notes Database - University of Applied Sciences, Ostwestfalen-Lippe, Lemgo: UC Irvine, 2012.