Segmentation of Text and Non-text in On-Line Handwritten Patient Record Based on Spatio-Temporal Analysis

Rattapoom Waranusast, Peter Haddawy, and Matthew Dailey

Computer Science and Information Management program,
School of Engineering and Technology, Asian Institute of Technology,
P.O. Box 4, Klong Luang, Pathumthani 12120, Thailand
{Rattapoom.Waranusast,haddawy,mdailey}@ait.ac.th

Abstract. Note taking is a common way for physicians to collect information from their patients in medical inquiries and diagnoses. Many times, when describing the pathology in medical records, a physician also draws diagrams and/or anatomical sketches along with the free-text narratives. The ability to understand unstructured handwritten texts and drawings in patient record could lead to implementation of automated patient record systems with more natural interfaces than current highly structured systems. The first and crucial step in automated processing of free-hand medical records is to segment the record into handwritten text and drawings, so that appropriate recognizers can be applied to different regions. This paper presents novel algorithms that separate text from non-text strokes in an on-line handwritten patient record. The algorithm is based on analyses of spatio-temporal graphs extracted from an on-line patient record and support vector machine (SVM) classification. Experiments demonstrate that the proposed approach is effective and robust.

Keywords: Automated patient record, Document segmentation, Spatio-temporal analysis, Online handwritten document.

1 Introduction

Introduction of computer-based patient record systems has become an important initiative in many countries in an attempt to improve healthcare quality and control costs [1]. While electronic patient record systems have many advantages over traditional paper based systems, acceptance among physicians has been slow. Many physicians find that electronic patient record systems are difficult to use require more time than traditional paper based systems [2, 3]. Electronic patient records are typically highly structured and make use of standardized vocabularies. While this facilitates processing of the information and communication among physicians, it has negative implications for the naturalness of the interaction. Physicians commonly use narrative and sketches to record patient information but the structured form of the records does not

C. Combi, Y. Shahar, and A. Abu-Hanna (Eds.): AIME 2009, LNAI 5651, pp. 345–354, 2009.

allow for this. Furthermore, physicians must map their concepts corresponding to their findings, diagnoses, and tests into the computer's predefined concepts, which requires time and can be constraining if the provided vocabulary is not sufficiently rich. All of these factors can result in the electronic patient record system interfering with the physician's though process rather than supporting it [4] [5]. Because of these factors, paper records still play a critical part in daily clinical work.

With the pen-based computing technology, the authors and colleagues are currently developing an electronic patient record system which can understand freehand anatomical sketches, annotations, and handwritten free-text. Understanding means that the system has the ability to recognize and interpret handwritten text, sketched drawings, and annotated symbols, and the ability to identify the context of the whole page. Such a system would combine the flexibility and user-friendliness of paper-based records with the ability to electronically process and search the information.

Normally, physicians use medical narratives in the form of unstructured text which are flexible, expressive, and familiar to them as an essential tool for diagnosis and decision making. They also often draw diagrams and/or anatomical sketches and annotate them. Some typical patient records are shown in Fig 1.

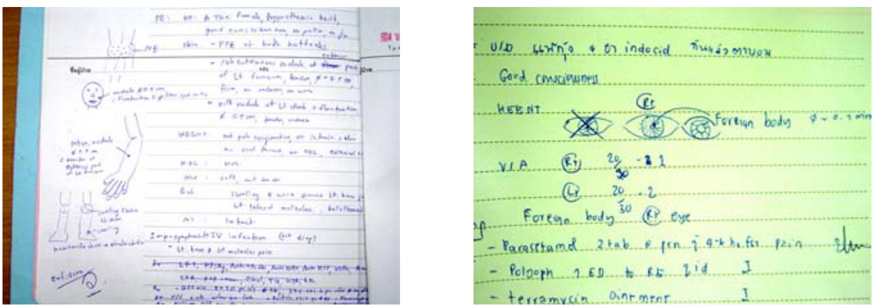

Fig. 1. Examples of paper-based patient records[1]

Because the page is potentially composed of different kinds of elements, the first and crucial step is to decompose the page into text and drawing regions. This step aims to group ink strokes of the same kind and to send them to proper recognizers in later steps.

In this paper, we present a novel approach to classifying ink strokes as text or non-text based on analyses of what we call spatio-temporal graphs. Instead of extracting features directly from ink stroke points as commonly found on other work in this field, features of the spatio-temporal graphs are calculated and extracted. The temporal neighborhoods are also taken into account based on the assumption that words and drawings are typically composed of contiguous strokes. We use a support vector machine (SVM) as a classifier to classify the strokes as text or non-text. The approach is robust to cursive and block hand writing.

[1] We thank to Dr. Siriwan Suebnukarn from Thammasat University for generously providing her medical expertise, data, and time.

2 Related Work

Since the late 1990's, the area of pen-based user interfaces has been very active. Applications include engineering drawings and simulations [6-8], architecture design [9], computer-aided design [10, 11], user interface and software design [12, 13], military planning [14], knowledge acquisition [15], music editing [16], and image retrieval [17, 18]. The only work in medical application of sketch-based interfaces that we know of is UNAS (UNderstanding Anatomical Sketches) [19], which is applied to COMET, a collaborative intelligent tutoring system for medical problem-based learning [20, 21].

In pen-based computing research, one of the most challenging problems is to separate the ink strokes into text, diagrams, and symbols. The objective of this process is to be able to cluster the strokes together and to send them to the right recognizers. This segmentation is necessary to design a robust interpretation of the ink even for an intelligent ink editing system [22]. Shilman et al. [23] also found that users prefer not to be constrained by facilitate ink understanding, such as pressing a button to switch modes or some special regions to identify the type of strokes [22]. Jain et al. [24] used a hierarchical approach to analyze on-line documents. They separate text and non-text stroke using only two stroke features, stroke length and stroke curvature, computed from individual strokes. This is based on the assumption that text strokes are typically short. However, this assumption cannot be applied to cursive writings and scribbles, which are ubiquitous in medical documents. Bishop et al. [25] proved that considering contexts of strokes allows better accuracy than using only stroke features. They use both features of the strokes and features of the gaps between strokes combined with temporal information using Hidden Markov model (HMM). Shilman et al. [23] use a completely different approach. They applied the bottom-up approach to separate text and graphics, that is starting with the strokes and repeatedly group them into letters, words, lines, and paragraphs. This method greatly depends on character recognition algorithms, which are not suitable for handwritten text scrawled by some physicians.

3 The System

The input data for our system are on-line digital ink documents, which are composed of sequences of strokes, separated by gaps. A stroke consists of a sequence of stroke points (x and y coordinates) recorded between a pen-down event (when the tip of the pen touch the screen) and a pen-up event (when the pen is lifted away from the screen). In addition to the spatial data of each stroke point, a time stamp indicating the time when the point was created is also recorded. These time stamps provide us the temporal ordering of the stroke points. It is obvious from other work [24-26] that the spatial information alone can give us rich useful features to separate text from non-text, we would expect to achieve better performance if we take temporal information in to account. We also believe that, people intuitively make the same kind of strokes consecutively before change to another kind of strokes. In other words, we can say that people will typically draw several graphics strokes in succession in order to draw a picture, or will scribble several text strokes in succession while writing some words. Because of this fact, we would also expect to improve the performance of the system by extracting information from a group of neighboring strokes instead of an isolation of individual stroke. We describe our approach in detail in the following subsections.

3.1 Spatio-Temporal Graphs

To extract information from both spatial and temporal aspects of an ink document, we construct two graphs representing spatio-temporal relationships of strokes in the document. An X-T spatio-temporal graph is a graph with its horizontal axis represents temporal information of strokes (time stamp of each stroke point) and its vertical axis represents x-component of spatial information (x coordinates of each stroke points), constructed by plotting x value against time-stamp of each stroke point. A Y-T spatio-temporal graph also has temporal information of strokes as its horizontal axis and y-component of spatial information as its vertical axis. The Y-T graph is constructed in the same manner as the X-T graph, which is plotting y coordinates of a stroke point against its time stamp.

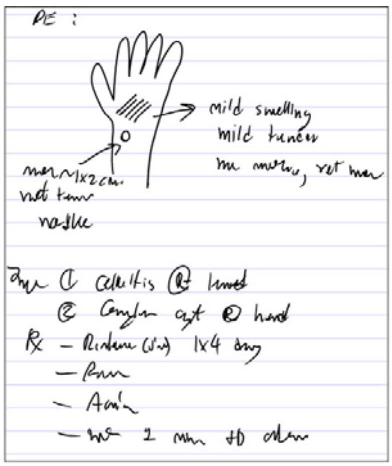

Fig. 2. A sample page of on-line patient records

Fig 2 shows a page of an on-line patient record, while Fig 3a and Fig 3b are the X-T and Y-T spatio-temporal graphs of the page in Fig 2, respectively. This patient record was written from left-to-right and top-to-bottom manner as found in the most western languages. In this example, the sketch of a patient's hand was drawn early in the page, which is corresponding to a part of the Y-T graph that look like a big U-shaped on the left side of the graph. We can notice from both X-T and Y-T graphs that there are some consistencies in the properties of text line strokes. We might say that stroke points of a text line, when plotting in X-T graph look like a slanted line, with its slope represents a horizontal speed of the writing. The fact that text letters have limited and almost the same height reflects as groups of horizontal lines in the Y-T graph. When we write a line of text, we move the pen much more up-and-downs than when we draw a picture. This fact is also shown as high frequency component

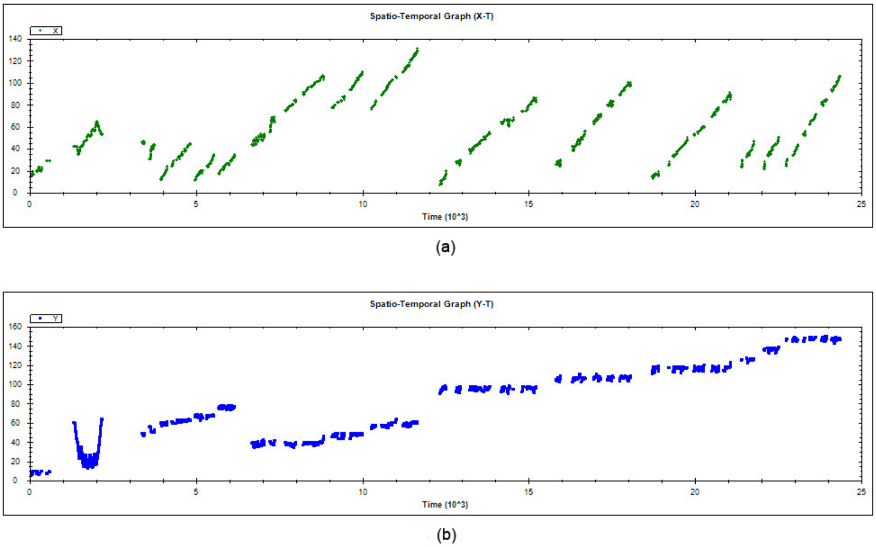

Fig. 3. Spatio-temporal graphs, X-T (a) and Y-T (b), constructed from the page in Fig 2

(looks like jagged line) in the graphs, especially the Y-T graph. From these unique properties of text lines in the spatio-temporal graphs, we can extract features from the graphs and use them to classify text or non-text strokes.

3.2 Features Extraction

At this point, we have two spatio-temporal graphs representing the whole strokes in the document, and we also know that text lines have certain unique properties, which are;

1) a text stroke and its neighbors form a line with certain angles in the X-T graph,
2) they form a horizontal line in the Y-T graph,
3) the widths of these lines (in both X-T and Y-T graphs) are consistency and limited, and
4) these lines are jagged with relatively high frequency.

We derive 9 features from these properties, which can be categorized into two groups, line-based, and frequency-based features. The followings are detail of these features.

Line-based Features: To compute the line-based features, we fitted an interesting graph point and its neighbors with a line. We use RANSAC [27] as a line-fitting method, because of its robustness to outliers. Once we fit a line to the graph points, we can compute; 1) An angle between temporal axis and fitted line in the X-T graph, 2) An angle between temporal axis and fitted line in the Y-T graph. 3) a correlation coefficiency of points in the X-T graph, 4) a correlation coefficiency of points in the

Y-T graph, 5) a standard error of estimation of the fitted line in the X-T graph, and 6) a standard error of estimation of the fitted line in the Y-T graph.

Frequency-based Features: In this category, we use two properties in computing the features. The first property is the relative extrema density [28], which is a measure for texture analysis. The relative extrema density are defined as the number of extrema per a period of time. The extrema are defined along the temporal axis in the following way. A graph point at time t is a relative minimum if its value f(t) satisfies

$$f(t) \le f(t+1) \quad \text{and} \quad f(t) \le f(t-1) \tag{1}$$

A graph point at time t is a relative maximum if

$$f(t) \ge f(t+1) \quad \text{and} \quad f(t) \ge f(t-1) \tag{2}$$

Note that with this definition, each point in the interior of any constant value run of points is considered simultaneously relative maxima. That is every points on a flat line or a plateau both are considered relative maxima. Plateaus at the local extrema, and plateaus on the way down from or up to the extremum also fall into this scenario, which is not desirable in our algorithm. To avoid this problem, we perform a preprocessing step and slightly modify the original relative extrema density algorithm. In the preprocessing step, we compress the graphs by eliminating the consecutive redundant value. To do that, the center of a plateau is considered to be a representative of that plateau, while the rest points in the plateau are ignored.

The second property is the stroke cusp density, which is the only property we compute directly from the stroke points, not the graph points. Stroke cusps are stroke points where the direction of the stroke has changed abruptly. The endpoints of each stroke are also considered as cusps. Fig 4 shows an example of cusps (depicted as superimposed red circles) on a sketch. From the figure, we can see that cusps are quite denser in a line of text (annotating words) than in a drawn picture (an eye). We define a stroke cusp density as the number of cusp in a period of time. We compute stroke cusps directly from the sketch, count them, and calculate the density.

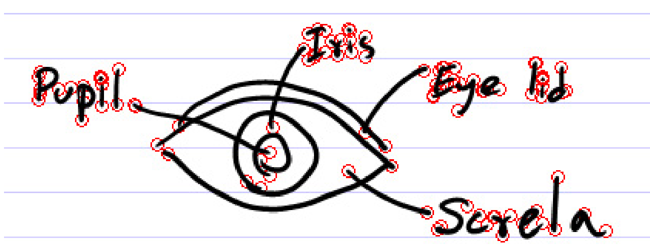

Fig. 4. An example of a sketch with cusps. Cusps are illustrated as red circles and superimposed on the sketch.

From the above definition of the relative extrema density and the cusp density, we derived three features from them as 7) a relative extrema density of the X-T graph, 8) a relative extrema density of the Y-T graph, and 9) a stroke cusp density.

To extract these 9 features, firstly, we have to cluster graph points into groups, because all of the above features can be computed only from a group of points. One possible way to compute these features is to divide a graph into some smaller segments, and extract these features from each segment. We have tried this approach, and found that this approach depends too much on divider algorithms. If the divider fails to divide a graph at proper positions, especially at the points between text and non-text graph points, the misclassifications are very high. To avoid this problem, we use moving window method to compute these features. The moving window is move along the temporal axis in both graphs. At a particular point in the graph that the window moves to, maximum weight is added to the point at the center of the window. Weights are gradually reduced and added to points that far away from the center of the window. The features at a particular point are computed from these weighted points.

At each stroke, features associated with the stroke are computed by averaging features of every stroke points in the stroke. The features of a stroke are kept to be used in the classification step.

3.3 Classification

Once all stroke features are computed, we construct a stroke classifier, which classifies strokes as text or non-text. Given these features, we believe that most standard classifiers can give us good results. We initially used k-nearest neighborhood (KNN) as our classifier. Then we tried a support vector machine (SVM) with C-SVM model and radial basis function to perform the classification. With this classifier, we performed parameter optimizations and found that the SVM is slightly more accurate than KNN.

4 Experimental Results

The experimental data was randomly collected from patient records at Thammasat University Hospital, Thailand. In total, 10,694 strokes (9,070 text strokes, and 1,624 non-text strokes) were collected from 83 patient records from approximately 8-10 physicians. Because the hospital used paper-based patient record and the physicians' time for their patients was very limited, it was unable to make the physicians sketch directly on a tablet PC. Thus we collected data by taking photos of the records at the end of examination sessions. Then, we copied the records into ink documents by displaying a patient record image on the tablet screen and sketching over it. Due to the limited number of data, we adopted 10-fold cross-validation technique to train and test the data. All strokes were stratified based on stroke class to ensure that the random sampling in cross-validation technique was done with each stroke class was properly represented in both training and test sets. To do this we employed Weka [29] as our test platform. Our algorithms achieved the accuracy of 94.61% in the 10-fold cross-validation test. Table 1 shows the confusion matrix of the results.

We also manually evaluated the results by looked through each classified page. We found that our approach provides good results, even though the text was scrawled and highly illegible. Fig 5 shows an example of our classified page, with non-text strokes marked in blue and text strokes in black.

Table 1. Confusion matrix showing the results of the evaluation. The rows correspond to the true class and the columns to the predicted classes.

		Classified as	
		Text	Non-Text
Actual	Text	8,938	132
	Non-Text	444	1,180

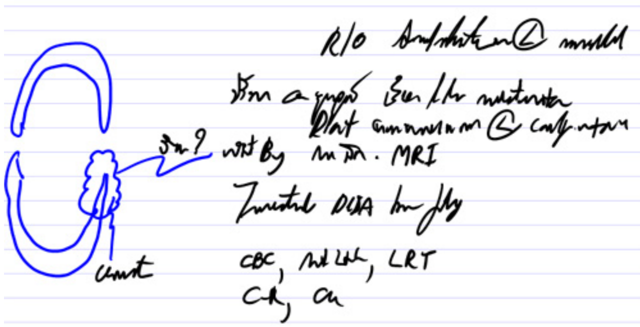

Fig. 5. An example of correctly classified scribbles in a patient record

5 Conclusion and Future Work

In this paper, we have presented a system for segmenting on-line patient records into text and non-text strokes. The system exploits features of the spatio-temporal graphs constructed from the patient record pages. The temporal neighborhoods of the spatio-temporal graph point are also taken into account, which based on the assumption that strokes of the same type tend to be correlated. We use support vector machine (SVM) as a classifier to classify the strokes. The experimental results demonstrate that the approach achieves high accuracy and is robust to cursive and scrawl writings which are ubiquitous in medical documents.

Although the accuracy of the approach is high, we believe that our current method can most probably be refined to yield better results. For example, at the moment, we ignore the properties of an individual stroke, which are usually utilized by most of other work in this area. For the moving window technique we applied in the algorithm, we move the window along the temporal axis of the graph. However, intuitively the strokes which are not in close spatial proximity should not be considered as the same group. Therefore, taking spatial axis into account in moving the window might improve the results. We also wish to gather more test data, especially on drawings, which will provide us more quantitative and qualitative insight of the nature of the medical documents.

References

1. HIMSS, 19th Annual HIMSS Leadership Survey: CIO Results Final Report. Healthcare Information and Management Systems Society (HIMSS) (2008)
2. Fraser, H.S., Biondich, P., Moodley, D., Choi, S., Mamlin, B.W., Szolovits, P.: Implementing Electronic Medical Record Systems in Developing Countries. Inform. Prim. Care. 13, 83–95 (2005)
3. Connolly, C.: Cedars-Sinai Doctors Cling to Pen and Paper. The Washington Post, March 21 (2005)
4. Walsh, S.H.: The Clinician's Perspective on Electronic Health Records and How They Can Affect Patient Care. Bmj. 328, 1184–1187 (2004)
5. Miller, R.H., Sim, I.: Physicians' Use of Electronic Medical Records: Barriers and Solutions. Health Affairs 23 (2004)
6. Stahovich, T.F., Interpreting the Engineer's Sketch: A Picture is Worth a Thousand Constraints. In: 1997 AAAI Symposium on Reasoning with Diagrammatic Representations II, Cambridge, Massachusetts, pp. 31–38 (1997)
7. Alvarado, C., Davis, R.: Intelligent Mechanical Engineering Design Environment: From Sketching to Simulation. MIT AI Laboratory Annual Abstracts. MIT, Cambridge, MA (2000)
8. Kara, L.B., Gennari, L., Stahovich, T.F.: A Sketch-based Tool for Analyzing Vibratory Mechanical Systems. Journal of Mechanical Design 130, (2008)
9. Leclercq, P.: Interpretative Tool for Architectural Sketches. In: 1st International Roundtable Conference on Visual and Spatial Reasoning in Design: Computational and Cognitive Approaches, Cambridge, MA, USA (1999)
10. Gross, M.D., Do, E.: Demonstrating the Electronic Cocktail Napkin: A Paper-like Interface for Early Design. In: Conference on Human Factors in Computing Systems, Vancouver, British Columbia, Canada, pp. 5–6 (1996)
11. Qian, D., Gross, M.: Collaborative Design with Netdraw. In: Computer Aided Architectural Design Futures Futures 1999, pp. 213–226 (1999)
12. Landay, J.A.: Interactive Sketching for the Early Stages of User Interface Design. Computer Science Dept., Vol. Ph.D. Carnegie Mellon University, Pittsburgh, Pa (1996)
13. Hammond, T., Davis, R.: Tahuti: A Sketch Recognition System for UML Class Diagrams - Extended Abstract. In: AAAI Spring Symposium on Sketch Understanding, pp. 59–68. AAAI Press, Stanford (2002)
14. Forbus, K.D., Usher, J., Chapman, V.: Sketching for Military Courses of Action Diagrams. In: 8th International Conference on Intelligent User Interfaces, pp. 61–68. ACM Press, Miami (2003)
15. Forbus, K., Usher, J.: Sketching for Knowledge Capture: A Progress Report. In: 7th International Conference on Intelligent User Interfaces, pp. 71–77. ACM Press, New York (2002)
16. Macé, S., Anquetil, E., Bossis, B.: Pen-Based Interaction for Intuitive Music Composition and Editing. In: Shen, J., Shepherd, J., Cui, B., Liu, L. (eds.) Intelligent Music Information Systems: Tools and Methodologies, pp. 261–288. Idea Group, USA (2007)
17. Fonseca, M., Barroso, B., Ribeiro, P., Jorge, J.: Sketch-Based Retrieval of ClipArt Drawings. In: Advanced Visual Interfaces (AVI 2004). ACM Press, Gallipoli (2004)
18. Leung, W.H., Chen, T.: Trademark Retrieval Using Contour-Skeleton Stroke Classification. In: IEEE Intl. Conf. on Multimedia and Expo (ICME 2002), Lausanne, Switzerland (2002)

19. Haddawy, P., Dailey, M., Kaewruen, P., Sarakhette, N.: Anatomical Sketch Understanding: Recognizing Explicit and Implicit Structure. In: Miksch, S., Hunter, J., Keravnou, E.T. (eds.) AIME 2005. LNCS, vol. 3581, pp. 343–352. Springer, Heidelberg (2005)
20. Suebnukarn, S., Haddawy, P.: A Collaborative Intelligent Tutoring System for Medical Problem-Based Learning. In: Int'l Conf. on Intelligent User Interfaces 2004, Madeira, Portugal, pp. 14–21 (2004)
21. Suebnukarn, S., Haddawy, P.: COMET: A Collaborative Tutoring System for Medical Problem-based Learning. IEEE Intelligent Systems 22, 70–77 (2007)
22. Anquetil, E., Lorette, G.: New Advances and New Challenges in On-line Handwriting Recognition & Electronic Ink Management. In: Chaudhuri, B.B. (ed.) Digital Document Processing: Major Directions and Recent Advances (Advances in Pattern Recognition), pp. 143–164. Springer, Heidelberg (2006)
23. Shilman, M., Wei, Z., Raghupathy, S., Simard, P., Jones, D.: Discerning Structure from Freeform Handwritten Notes. In: Seventh International Conference on Document Analysis and Recognition. IEEE Computer Society, Los Alamitos (2003)
24. Jain, A.K., Namboodiri, A.M., Subrahmonia, J.: Structure in On-line Documents. In: ICDAR 2001, pp. 844–848 (2001)
25. Bishop, C.M., Svensen, M., Hinton, G.E.: Distinguishing Text from Graphics in On-Line Handwritten Ink. In: 9th International Workshop on Frontiers in Handwriting Recognition (IWFHR 2004). IEEE Computer Society, Los Alamitos (2004)
26. Namboodiri, A.M., Jain, A.K.: Robust Segmentation of Unconstrained Online Handwritten Documents. In: ICVGIP, pp. 165–170 (2004)
27. Fischler, M.A., Bolles, R.C.: Random Sample Consensus: A Paradigm for Model Fitting with Applications to Image Analysis and Automated Cartography. Comm. of the ACM 24, 381–395 (1981)
28. Rosenfeld, A., Troy, E.B.: Visual Texture Analysis. In: Conf. Rec. Symp. Feature Extraction and Selection in Pattern Recognition, p. 115 (1970)
29. Witten, I.H., Frank, E.: Data Mining: Practical Machine Learning Tools and Techniques, 2nd edn. Morgan Kaufmann, San Francisco (2005)