Identifying Word Boundaries in Handwritten Text

Yi Sun, Timothy S. Butler, Alex Shafarenko, Rod Adams, Martin Loomes, Neil Davey
Department of Computer Science
Faulty of Engineering and Information Sciences
University of Hertfordshire
College Lane, Hatfield
Hertfordshire AL10 9AB
Y.2.Sun, t.s.butler, a.shafarenko, r.g.adams, m.j.loomes, N.Davey@herts.ac.uk

Abstract

Recent work on extracting features of gaps in handwritten text allows a classification of these gaps into **inter-word** *and* **intra-word** *classes using suitable classification techniques. In the previous work, we apply 5 different supervised classification algorithms from the machine learning field on both the original gap dataset and the gap dataset with the best features selected using mutual information. In this paper, we improve the classification result with the aid of a set of feature variables of strokes preceding and following each gap. The best classification result attained suggests that the technique we employ is particularly suitable for digital ink manipulation at the level of words.*

1. Introduction

In this paper, we further address the problem of identifying word boundaries in handwritten text: a process known as word segmentation. We make use of a selection of contemporary classification algorithms, such as multi-layer perceptrons, support vector machines, and Gaussian mixture models.

In [9] we tried to find a suitable classifier to automatically segment so-called *digital ink*: graphically enhanced fragments of pen trace representing handwritten words, shapes and symbols, of the sort that usually appear on paper when real ink is used for writing. Further details about the problem domain can be found in the next section. The previous work was done by applying classifiers using features of *gaps* between adjacent pen strokes. Here we attempt to improve the performance of the classifier by including features of these 2 strokes as well as the gap itself.

In this work, we first produce a new dataset using stroke and gap information. We then test 5 different supervised classification learning algorithms from the machine learning field to categorise gaps.

We expound the problem domain in the next section. In Section 3, we introduce the datasets used in this paper. All experimental results are given in Section 4. The paper ends in Section 5 with a discussion.

2. Problem domain

In this paper we focus on one level of the semantic penetration of pen input: the level of words. More detail can be found in [9]. By 'word' we mean a group of pen strokes that have lexical significance, i.e. one that represents a word in a human language or a distinct symbol that can be used as a word. We wish to automatically segment digital ink represented as a *sampled pen trace* into word fragments purely on the basis of spatiotemporal relations between consecutive strokes, ignoring any meaning that may be represented by each such stroke. This has been a known problem in handwriting recognition research as well, although in this area of technology, word segmentation is seen merely as a precursor to full character recognition.

To extract features, we have been guided by [7] where a thorough geometric and temporal features were provided for a pen gesture recogniser. We illustrate some of features in Figure 1. It presents a single pen stroke with its bounding box. The features x and y as shown give the dimensions of the bounding box and the angle α is linked with its aspect ratio. The distance s is between the end points of the stroke, and β is the angle between the line connecting those points and the vertical. Finally, if θ_i is the angle between two consecutive pen segments of the stroke, i and $i+1$, then one can use the feature $\sigma = \sum_{i=1}^{n-1} \theta_i$ as a measure of curvature. The proposed features included a few related to the time interval of the stroke and the speed of the pen tip as well. We have introduced a gap feature which has proven especially useful for our purposes. We call it *river width* or *river* for short, following Fox and Tappert [5]. The river of a gap is the shortest distance between two consecutive strokes, i.e. the length of the shortest chord drawn between pen position samples from neighbouring strokes, as shown in Figure 2. Two rivers are indicated there by double-headed arrows.

We have expanded the set proposed in [5] by our own form factors, see [4], for each pen stroke. The pen trace has thus been abstracted to a sequence of strokes and gaps, where each gap is represented by 14 feature variables,

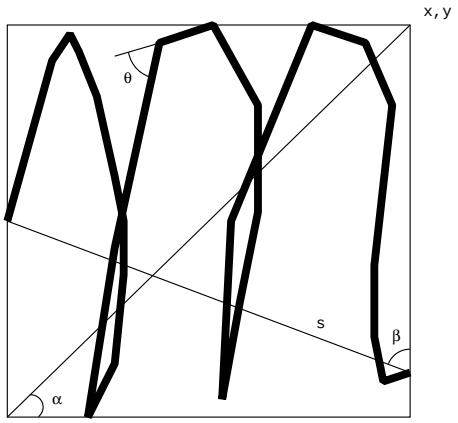

Figure 1. **An illustration: the sort of features of a single pen stroke with its bounding box.**

Figure 2. **An illustration: two rivers of gaps are shown by double-headed arrows.**

while each stroke by 25. In this work, we are interested in classifying gaps. A human reader has annotated the gaps in our experimental traces as either intra-word or inter-word by recognising the words in the language. Thus the task is to search for a classification method which can produce the same annotations with as few errors as possible.

3. The description of the datasets

The *original* gap dataset includes 2482 data points labeled by *inter-word* and 4980 *intra-word*. In [9], we presented experimental results with all 14 features and *reduced* features involving the 8 most significant to the classification found by analysing mutual information. Figure 3 shows the mutual information of each feature with the class variable sorted by their values. As shown, there is a reasonable "jump" from the ninth value to the eighth. We ignore those features indexed from 9 to 14. Thus 8 features with mutual information values more than 0.3, a subset of all of the features, can be obtained. More detail of applying mutual information for feature extraction can be found in [9].

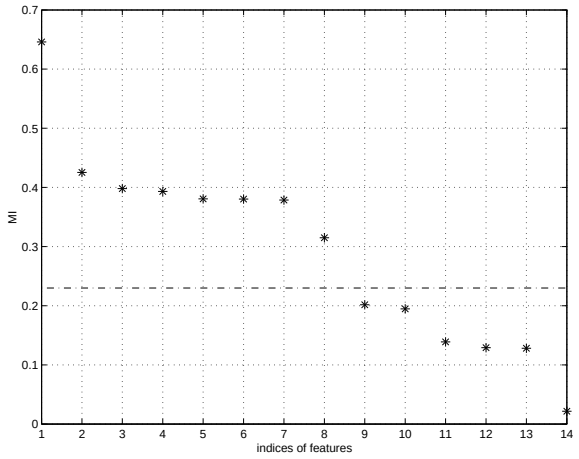

Figure 3. **Mutual information of class variable and each feature of gaps: each value is shown as a star sign. The background are dashed lines that are major grid lines to the current axes. The horizontal dash-dot line denotes the cut off value.**

In previous work, we were able to correctly classify nearly a half of the data to a 99% accuracy from the value of the *river* feature alone, which presented in Figure 3 is the most significant to classification. Since it measures the shortest distance between samples in adjacent strokes, gaps between words usually have a larger value than gaps within words. One can expect to benefit from this variable as much as possible, though exceptions often occur with variety in writing styles as mentioned in the introduction section. Specifically those gaps with a *river* value above a particular threshold value are extremely likely to be inter word gaps, and those below a lower threshold are likely to be intra word gaps; such vectors are denoted as *evident* data. Two thresholds of the values of *river* can be determined as displayed in Figure 4. In this figure, the river values increase from left to right. Boundary 1 specifies a *river* value, on the left of which one can ensure that the probability that the gap belongs to class *intra-word* is not less than 99 percent; while boundary 2 specifies another value of *river*, on the right of which the probability that the gap belongs to class *inter-word* is not less than 99 percent. Then the whole dataset is filtered by means of these two thresholds. In this way, a sub-dataset called *hard*, whose values of the *river* feature are within these two boundaries, is obtained. This subset therefore consists of 3361 gaps that cannot easily be classified by the *river* feature.

3.1. The hard stroke-gap-stroke dataset

One might expect that a further improvement in classification could be achieved by utilising more information from the characteristics of the preceding and following strokes of a gap. To this end, a new set of vectors was created by concatenating a gap with its preceding stroke and its

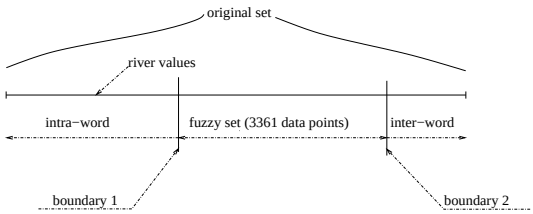

Figure 4. **The fuzzy dataset is generated with two boundaries.**

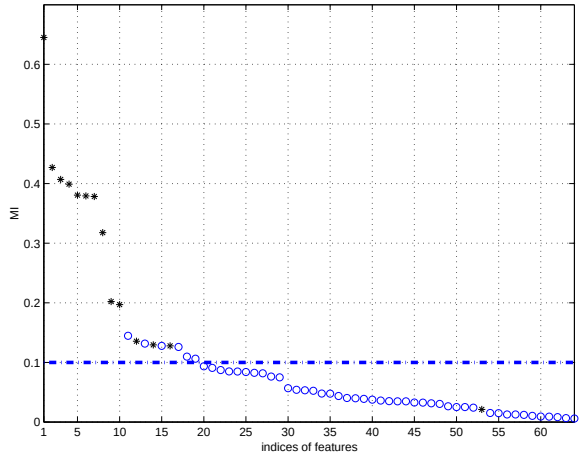

Figure 5. **Mutual information of the class variable and each feature: each value is shown as a star sign if the corresponding feature is one of gap variables, otherwise it is presented as a circle if the feature is one of stroke variables. The horizontal dash-dot line denotes the cut off value.**

following stroke, i.e. stroke-gap-stroke. This dataset contains the same number of data points as the original gap dataset, namely 7462 points. However, this set now has a total of 64 features, for each stroke we have 25 additional features, and is called *SGS-64*. As before, one can apply mutual information to select a subset of features for this dataset [9].

It can be seen that the 10 most significant features for classification are in fact all gap feature variables. Since we want to know whether the combination of stroke and gap features can improve the categorisation, a cut off value of 0.1 is used so that a set of stroke features can be involved. This produces a reduced dataset, called *SGS-19*, with the same number of data points but only 19 features. Using the *river* feature, which is still the most significant feature in Figure 5, one can again produce a *hard* subset from this reduced feature dataset, called *Hard-SGS-19*.

Table 1. **19 significant features selected using mutual information.**

No.	Feature	From
1	River	gap
2	X-displacement	gap
3	Distance between centre of gravity	gap
4	Displacement	gap
5	Length of the bounding box diagonal	gap
6	Num. of samples in stroke	gap
7	Duration of stroke	gap
8	Distance between first and last point	gap
9	Cosine between first and last point	gap
10	Distance between centre of gravity incl. pressure	gap
11	Length of the bounding box diagonal	following stroke
12	Angle of the bounding box diagonal	gap
13	Sine between first and last point	following stroke
14	Sine between first and last point	gap
15	Num. of samples in stroke	following stroke
16	Y-displacement	gap
17	Duration of stroke	following stroke
18	Total stroke length	following stroke
19	Total stroke length	preceding stroke

Table 1. lists 19 selected features using mutual information. It includes all but one gap features and 6 new stroke features. Interestingly only 1 feature is selected by mutual information from the stroke preceding the gap. Note that some gap features, such as No. 5 and 6, are the same as those used for describing a stroke. During the data collection procedure, pen data is recorded while the pen is within 5mm of the table surface. While pen pressure is under a certain threshold, the pen is considered to off the tablet surface (i.e. a gap). However, there will be some pen-trace information describing the movement of the pen over a gap. At the very minimum this will include the start and end of the gap as the pen moves out of and in to proximity. At the most it will describe the movement of the pen over the entire gap. Table 2. shows brief explainations of these 19 features.

4. Experimental results with supervised classifiers

4.1. Supervised classifiers

In this section, we first list the supervised classifiers used in our experiments. Readers who are interested in those

Table 2. **Descriptions of 19 significant features.**

No.	Feature	Description
1	**River**	This is the minimum distance between two strokes. Distances are calculated between all the sample points in stroke N and those in stroke $N+1$.
2	**X-displacement**	This is the $\mathbf{X}$-displacement between the end of stroke N and the start of stroke $N+1$ (start and end of the gap).
3	**Distance between centre of gravity**	The centre-of gravity of a stroke is the average all the sample points: $\mathbf{x}$ = avg. of all $\mathbf{x}$ coords, $\mathbf{y}$ = avg. of all $\mathbf{y}$ coords.
4	**Displacement**	As $\mathbf{X}$-displacement, but in 2 dimensions ($\mathbf{X}$ and $\mathbf{Y}$ coords).
5	**Length of the bounding box diagonal**	A bounding box is found round the coords of the sample points for a stroke. Minimum and maximum $\mathbf{X}$ and $\mathbf{Y}$ coords are found by checking each sample point in the stroke. The length of the diagonal of this box is then found.
6	**Num. of samples in stroke**	The number of sample points in a stroke.
7	**Duration of stroke**	The time taken to draw (write) the stroke. All sample points are time stamped, so this is the Time(end) − Time(start).
8	**Distance between first and last point**	The distance between $\mathbf{X}$, $\mathbf{Y}$ of start point and $\mathbf{X}$, $\mathbf{Y}$ of the end point.
9	**Cosine between first and last point**	This is the cosine of the angle between the last $\mathbf{X}$, $\mathbf{Y}$ sample point of stroke $N+1$ and first $\mathbf{X}$, $\mathbf{Y}$ sample point of stroke N
10	**Distance between centre of gravity incl. pressure**	This is as **Distance between centre of gravity**, but is in a 3-dimensional space, with pen pressure constituting the thrid dimension.
11	**Length of the bounding box diagonal**	As **Length of the bounding box diagonal** above
12	**Angle of the bounding box diagonal**	This is the angle between minimum and maximum $\mathbf{X}$ and $\mathbf{Y}$ coords as found in **Length of the bounding box diagonal** above.
13	**Sine between first and last point**	This is the sine of the angle between the first $\mathbf{X}$, $\mathbf{Y}$ sample point of stroke N and last $\mathbf{X}$, $\mathbf{Y}$ sample point of stroke $N+1$
14	**Sine between first and last point**	This is the sine of the angle between the last $\mathbf{X}$, $\mathbf{Y}$ sample point of stroke $N+1$ and first $\mathbf{X}$, $\mathbf{Y}$ sample point of stroke N.
15	**Num. of samples in stroke**	As (6) above.
16	**Y-displacement**	As (2) above, but in the $\mathbf{Y}$-displacement.
17	**Duration of stroke**	As (7) above.
18	**Total stroke length**	The sum of the distances between adjacent sample points along a stroke.
19	**Total stroke length**	As (18) above.

classification techniques can follow the references to learn more.

- Logistic discrimination analysis (LDA) [2];
- K-nearest neighbor classification (KNN) [6];
- Guassian mixture model (GMM) [2];
- Multi-layer perceptron (MLP) using scaled conjugate gradients algorithm [2];
- Support vector machine (SVM) using Gaussian kernel [8].

Parameters of each class-condition density were estimated from the training dataset in the GMM. For the MLP, a two-layer architecture was set up, since it has been proved for classification tasks that the MLP with sigmoidal activation function and two layers of weights can approximate any decision boundary to arbitrary accuracy [3].

4.2. Experiments

Experiments were performed on the hard dataset with all 14 gap features, the selected set of 8 gap features, and 19 stroke-gap-stroke features, namely, hard-G-14, Hard-G-8 and hard-SGS-19. In the experiments, $2/3$ of the data points from the dataset are used for training, while $1/3$ are used for testing. The user-chosen parameters for each classifier were selected by cross-validation, where the training set was divided into 10 partitions. 9 partitions were used to train the model and the other one was used as a validation set. In Table 3., we present all the user-chosen parameters attained by using cross-validation. The SVM experiments were completed using LIBSVM, which is available from the URL `http://www.csie.ntu.edu.tw/~cjlin/libsvm`. The others were implemented using the NETLAB toolbox, which is available from the URL `http://www.ncrg.aston.ac.uk/netlab/`.

Table 3. **User-chosen parameters from cross-validation. K denotes the number of neighbours; $nc1$ and $nc2$ are the number of Gaussian models in each mixture; j signifies the number of hidden units in the MLP; A is the upper bound of coefficients α_i in the SVM; and σ is width of radial basis function.**

	KNN (K)	GMM ($nc1$, $nc2$)	MLP (j)	SVM (A, σ^2)
Hard-G-8	9	6, 6	8	25, 0.16
Hard-G-14	9	6, 4	5	20, 0.1
Hard-SGS-19	9	9, 6	5	20, 0.1

4.3. Classification results

It can be seen in Figure 6 that for all but one of the classifiers the results for the SGS data are noticeable better

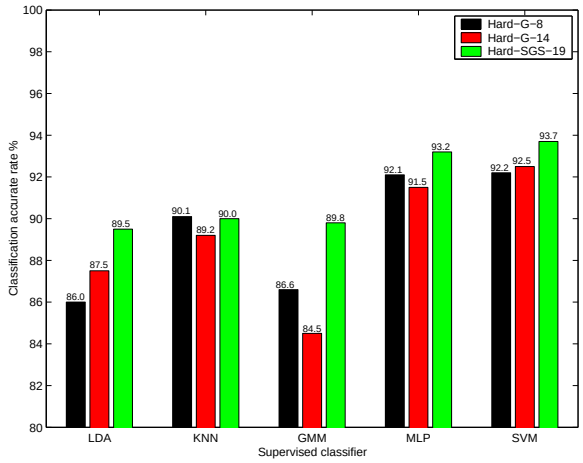

Figure 6. **Bar graph: classification results for each *hard* test dataset with 8, 14, 19 feature variables. The corresponding accuracy is shown on the top of each bar.**

than the gaps only data. In particular, the GMM performs much better on the Hard-SGS-19 dataset.

Furthermore, when considering the *hard* dataset with 19 feature variables, the SVM classifier gives a classification rate 93.7% which we can amalgamate with the *evident* data points to calculate a final classification rate of 96.6% for the whole dataset as shown in [9]. This is our best classification result.

5. Discussion

Our aim in this piece of work has been to investigate whether the performance of a word segmentation system could be improved using additional feature information. To do so, we took feature representations of the stroke preceding and following each gap. We analysed the mutual information of all the 64 resulting features with the two classes of gap. By taking the 19 best features we were able to include six stroke features. The classification results for the best classifiers, again the MLP and SVM, showed a notable reduction in the error rate - roughly one sixth of misclassifications were removed. Our best classifier, the SVM on the Hard-SGS-19 dataset, in combination with predicting on the basis of the *river* feature on the *evident* dataset gave an overall classification of unseen gaps at 96.6% correct.

Such a high level of word entity identification is likely to be sufficient to support digital ink applications in which character recognition is not used. Indeed the fact that word structure can be identified with less than one error in 30 words and irrespective of overall legibility of text, makes this technique especially suitable for digital ink manipulation at a whole word level. The cost of error (when errors are infrequent) is small: an occasional misclassification would only split a word in two resulting in a minor problem, e.g. a line break mid-word. For a small text area characteristic of Tablet PC and Personal Digital Assistants applications, this would happen once or twice in a screen, which would be completely acceptable. On the other hand, since writing on digital media is generally less easy than it is using pen and paper, some support for editing hand-written text, if only at a level of whole word manipulation, is crucial to ensure that stylus-based note-taking and document-processing are accepted by the mainstream user.

References

[1] R. Bellman, *Adaptive Control Processes: A Guided Tour*, Princeton University Press, New Jersey, 1961.

[2] C.M. Bishop, *Neural Networks for Pattern Recognition*, Oxford University Press, New York, 1995.

[3] E.K. Blum, L.K. Li, "Approximation Theory and Feedforward Networks", *Neural Networks*, Vol. 4, No. 4, 1991, pp. 511–515.

[4] T.S. Butler, *Human Interaction with Digital Ink: Legibility Measurement and Structural Analysis*, Ph.D. thesis, University of Hertfordshire, Department of Computer Science, University of Hertfordshire, Hertfordshire, UK, 2004.

[5] A.S. Fox, and C.C. Tappert, "On-line External Word Segmentation for Handwriting Recognition", *Proceedings of the Third International Symposium on Handwriting and Computer Applications*, 1987, pp. 53–55

[6] I.T. Nabney, *Netlab Algorithm for Pattern Recognition*, Springer, 2001.

[7] D. Rubine, "Specifying Gestures by Example", *Computer Graphics*, Vol. 25, No. 4, 1991, pp. 329-337

[8] B. Scholköpf, and A.J. Smola, *Learning with Kernels: Support Vector Machines, Regularization, Optimization, and Beyond*, The MIT Press, 2002.

[9] Y. Sun, T.S. Butler, A. Shafarenko, R. Adams, M. Loomes, and N. Davey, "Word Segmentation of Hndwritten Text Using Supervised Classification Tchniques", *Proceeding of IJCNN*, Budapest, 2004.