

9803634-6

IPOS's copy

1

"A method of recognising a table within a block of text" ✓

THIS INVENTION relates to a method of recognising a table within text and, more particularly to a method of recognising the location of a table within a block of text

Many texts contain tables. A table is a convenient form of conveying certain information such as numerical information and to allow swift comparison between values in a table. A text which includes a table will probably need to be processed or handled differently to text which does not include a table. For example, the text may need to be centre justified and the table may need to be formatted to provide grid lines or a border around the table. Thus, there exists a need to be able to identify a table within text and more specifically to identify a table comprising text subject-matter from a block of text.

US-A-5,668,892 discloses a table recognition apparatus which extracts information from a table within text with reference to the ruled lines bounding the table and the rows and columns within the table. This document teaches the use of a fixed deterministic algorithm to identify the presence of tables, which tables are commonly earmarked by dashed lines or other special designated characters. However, tables do not always appear in text in such a standard format and are extremely difficult to identify using the fixed deterministic algorithm disclosed in US-A-5, 668, 892.

Accordingly, it is an object of the present invention to seek to provide a method of locating a table within a block of text which does not use a fixed deterministic algorithm and which does not require the table to be earmarked by dashed lines or any other special designated characters.

Accordingly, one aspect of the present invention provides a method of recognising the location of a table within a block of text comprising the steps of: attributing a set of values to each line of a sample block of text, which set of values are determined at least by the characteristics of the line of text, the set of values for the sample block of text comprising training examples; indicating whether each line includes part of a table; attributing a set of values to each line of a block of text to be analysed, which set of values are determined at least by the characteristics of that line of text; applying a pattern recognition algorithm to each line of the block of text being analysed and lines from the training examples to identify whether the line of the block of text being analysed includes a part of a table.

Another aspect of the present invention provides a method of producing training examples of sample text to be used with a method of recognising the location of a table within a block of text, the method of producing training examples comprising the steps of: attributing a set of values to each line of a sample block of text, which set of values are determined at least by the characteristics of the line of text, the set of values for the sample block of text comprising training examples; and indicating whether each line includes part of a table.

In order that the present invention may be more readily understood, an embodiment thereof will now be described, by way of example, with reference to the accompanying drawings, in which:

Figure 1 is a text sample including a table;

Figure 2 comprises a set of values attributed to the respective lines of the text sample of Figure 1 in accordance with an embodiment of the present invention;

Figure 3 shows the set of values of Figure 2 annotated to indicate whether a given line is a part of a table;

Figure 4 is a further text sample including a table; and

Figure 5 comprises a set of values attributed to the respective lines of the text sample of Figure 4 in accordance with an embodiment of the present invention.

Referring to the drawings, the embodiment of the present invention to be described comprises a two-part process incorporating a learning process and then a subsequent recognition process.

The learning process involves the analysis of a number of sample texts each of which contains one or more tables. Each sample text comprises a number of lines of text. For the sake of ease of illustration, the following embodiments relate to lines of text which each comprise a string of 128 characters. It will be appreciated that the invention is equally applicable to lines of text with more or less characters and is not restricted to a fixed number of characters in a line.

As will be described below, each text line is attributed with a set of values which represent features or characteristics of at least that line but also, advantageously, features of the immediately preceding line and features of the immediately following line.

Once the respective values have been attributed to a line, each line is manually identified as either being a part of a table or not.

In the illustrated example, Figure 1 shows a sample block of text comprising 15 lines. A set of 27 feature values are assigned to each line as a function of nine features of a text line. The first nine feature values are assigned as a function of the immediately preceding line, the second nine feature values are assigned as a function of the present line and the third nine feature values are assigned as a function of the immediately following line.

For a given line, the nine features and their respective associated values are as follows:

F1: this feature encodes whether the line is a blank line having no text characters or a line having text characters. This feature is represented by a symbolic value of *l* if the line is a blank line or *f* if the line has any text characters.

F2: this feature encodes the number of leading spaces in the line before the first text character and is represented by a numerical value being the number of initial spaces before the first text character.

F3: this feature encodes the first character in the line that is not a space character. This feature is represented by one of the following special characters if the first character is one of those special characters: *()[]{}<>+~*/_=~/!@#%&* or, if the first text character is not one of the above special characters, then this feature is represented by the symbol *N*.

F4: this feature encodes the last character in the line that is not a space character. This feature is represented by one of the following special characters if the last character is one of those special characters: *()[]{}<>+~*/_=~/!@#%&*

`*/_~!@#$$%^&` or, if the last text character is not one of the above special characters, then this feature is represented by the symbol *N*.

F5: this feature encodes whether the line consists entirely of one special character only. Its possible values are either one of the following special characters: `()[]{}<>+-*/_~!@#$$%^&` or, if the line does not consist entirely of that special character, then this feature is represented by the symbol *N*.

F6: this feature encodes the number of segments in the line with two or more contiguous space characters. This feature is represented by the number of segments in the line with two or more contiguous space characters.

F7: this feature encodes the number of segments in the line with three or more contiguous space characters. This feature is represented by the number of segments in the line with three or more contiguous space characters.

F8: this feature encodes the number of segments in the line with two or more contiguous separator characters, where a separator character is one of the following: `" . * -`. This feature is represented by the number of such segments.

F9: this feature encodes the number of segments in the line with three or more contiguous separator characters, where a separator character is one of the following: `" . * -`. This feature is represented by the number of such segments.

As will be appreciated, features F1, F3, F4 and F5 are symbolic features whose values are one of a set of possible symbolic values whilst features F2, F6, F7, F8 and F9 are numerical features whose values are numeric integers.

Referring now to Figure 2, this shows the feature values of each of the 15 lines of the text sample of Figure 1. For example, let us consider the feature values attributed to line 2. The feature values for line 2 are as follows:

f,1,N,N,N,0,0,0,0,f,1,N,N,N,1,1,0,0,t,128,N,N,N,0,0,0,0

The first nine feature values - *f,1,N,N,N,0,0,0,0* - are derived from line 1, the next nine feature values - *f,1,N,N,N,1,1,0,0* - are derived from the present line, line 2, and the final nine features - *t,128,N,N,N,0,0,0,0* - are derived from the next line, line 3.

Analysing the nine feature values (the second nine feature values) of the present line, line 2, these values have been attributed as follows:

Feature Name	Feature Value	Explanation
F1	<i>f</i>	Line 2 is not blank so the symbolic value is <i>f</i> .
F2	<i>1</i>	The number of leading spaces in line 2 is one since there is a blank space before the character "A" in the first word "American".
F3	<i>N</i>	The first non-space character (<i>A</i>) is not one of the special characters <i>0 []{}<>+*/_~!@#\$%^&</i> so the symbolic value is <i>N</i> .
F4	<i>N</i>	The last character in line 2 which is not a space is "." So the symbolic value of F4 is <i>N</i> .
F5	<i>N</i>	Line 2 does not consist of one special character only so the value of F5 is <i>N</i> .
F6	<i>1</i>	After the 40th character in Line 2 ".", there are 88 contiguous space characters which qualify as a segment of two or more contiguous space characters so the value of F6 is <i>1</i> .
F7	<i>1</i>	After the 40th character in Line 2 ".", there are 88 contiguous space characters which qualify as a segment of three or more contiguous space characters so the value of F7 is <i>1</i> .
F8	<i>0</i>	There are no contiguous separator characters in line 2 (".*-") so the value of F8 is 0.
F9	<i>0</i>	There are no contiguous separator characters in line 2 (".*-") so the value of F9 is 0.

In the same manner, the table below analyses the second nine feature values of line 14, which values have been attributed as follows:

Feature Name	Feature Value	Explanation
F1	<i>f</i>	Line 14 is not blank so the symbolic value is <i>f</i> .
F2	3	The number of leading spaces in line 14 is three since there are three blank spaces before the character "Y" in the first word "Year".
F3	<i>N</i>	The first non-space character (Y) is not one of the special characters: 0 []()<>+,*/_=,!@#\$%^& so the symbolic value is <i>N</i> .
F4	%	The last character in line 14 which is not a space is "%" which is a special character so the symbolic value of F4 is %.
F5	<i>N</i>	Line 14 does not consist of one special character only so the value of F5 is <i>N</i> .
F6	3	There are three segments in Line 14 which have two or more contiguous space characters: 1) the three spaces before the word "Year"; 2) the three spaces between "15,029,000" and "66.9%" and 3) the 75 spaces after the last character "%".
F7	3	As F6, the numerical value is 3
F8	1	There is a contiguous string of 18 separator characters "." in line 14 so the value of F8 is 1.
F9	1	As F8, the numerical value is 1.

Turning from Figure 2 to Figure 3, the resultant feature values for each line are then manually annotated with a symbol to identify whether or not the corresponding line from the text sample shown in Figure 1 is part of a table or not. If the given line is part of a table, then the feature values are annotated with a plus sign (+) or if the given line is not part of a table, then the feature values for that line are annotated with a minus sign (-). This manual annotation indicates whether the line is part of a table or not.

The above described process represents the learning process which is the first stage in identifying the location of a table within a block of text. This first stage is important since it represents a transformation of input sample text into a form represented by feature values which is solvable by a pattern recognition algorithm such as a nearest neighbour algorithm. Without transforming the text

into feature values, it would not be possible to utilise pattern recognition algorithms in the following recognition stage. Thus, the present embodiment of the invention provides a solution to the problem of recognising the location of a table within a block of text using pattern recognition algorithms, a problem which has not previously been solved using pattern recognition algorithms.

The second stage in the process comprises the recognition of a table within a block of text.

Referring to Figure 4, this represents a further text sample having, in this case, 18 lines. The first step in the recognition process is to attribute each line with the 27 feature values described above so that the text information shown in Figure 4 can be manipulated using pattern recognition algorithms. Applying the same rules described above in relation to Figure 2, the 27 feature values for each of the 18 lines of the text sample shown in Figure 4 are set out in Figure 5.

Any pattern recognition algorithm can be used. In particular, a nearest neighbour algorithm can be applied in the recognition process to determine whether each of the lines of the text in Figure 4 is part of a table or not.

In the present example, this is illustrated using the nearest neighbour algorithm. This algorithm computes the distance $D(S, T)$ of the line in the text sample (T) to every training example from text sample (S). Thus, $D(S1, T1)$ is the distance measure between line 1 of training example text S from Figure 1 to line 1 of text sample T from Figure 4, S1 being the first line of training example S and T1 being the first line of text sample T. $D(S1, T1) = \sum_{j=1,27} d(s1_j, t1_j)$, where $d(s1_j, t1_j)$ is the distance between the j th feature value of S1 and T1.

As previously described, some of the feature values have symbolic values and others have numerical values. For numerical feature values, for example, taking the second feature value sl_2 of line 1 of text S (S1) and the second feature value tl_2 of line 1 of text T (T1), $d(sl_2, tl_2) = |(sl_2 - tl_2)/128| = |(128 - 128)/128| = 0$.

For symbolic values, $d(s, t)$ is either 0 or 1. If the symbolic value of the feature value in the training example is the same as the symbolic value of the feature value in the line being analysed, then $d(s, t) = 0$ but if the symbolic value of the feature value in the training sample is not the same as the symbolic value of the feature value in the line being analysed, then $d(s, t) = 1$. For example, taking the first feature value sl_1 of line 1 of text S (S1) and the first feature value tl_1 of line 1 of text T (T1), where, in this example, $sl_1 = "r"$ and $tl_1 = "t"$, then, $sl_1 \neq tl_1$ and thus $d(sl_1, tl_1) = 1$.

This process is repeated for each of the 27 feature values on line 1 of texts T and S to produce the following distances:

I	$d(sl_i, tl_i)$	I	$d(sl_i, tl_i)$	I	$d(sl_i, tl_i)$
1	0	10	0	19	1
2	0	11	0	20	0.9921875
3	0	12	0	21	0
4	0	13	0	22	0
5	0	14	0	23	0
6	0	15	0.0078125	24	0.0078125
7	0	16	0.0078125	25	0.0078125
8	0	17	0	26	0
9	0	18	0	27	0

The resultant values are then summed to produce the distance $D(S_1, T_1) = 2.0234375$ between the first line S_1 of training example text S and the first line T_1 of text sample T .

The same process is then repeated for every other line in the training example text S and the text sample T to produce distances $D(S_{1-15}, T_{1-18})$. The smallest distance or distances between the training sample S and the line (T_1 to T_{18}) being analysed is/are selected to make the decision as to whether the line being analysed comprises a line which is part of a table or not. The table below shows the distance values when comparing the first line of text T being analysed from Figure 4 to each of the lines of text in the training example text S shown in Figure 1:

$D(S_1, T_1)$	2.0234375
$D(S_2, T_1)$	1.9921875
$D(S_3, T_1)$	6.0234375
$D(S_4, T_1)$	0
$D(S_5, T_1)$	6.0234375
$D(S_6, T_1)$	0
$D(S_7, T_1)$	6.0234375
$D(S_8, T_1)$	0
$D(S_9, T_1)$	5.765625
$D(S_{10}, T_1)$	2.0546875
$D(S_{11}, T_1)$	5.1015625
$D(S_{12}, T_1)$	5.8671875
$D(S_{13}, T_1)$	7.15625
$D(S_{14}, T_1)$	7.1484375
$D(S_{15}, T_1)$	6.109375

As can be seen from the above table, the smallest distances occur at S4, S6 and S8. Since there are three matches, the first match is selected, i.e. S4.

Thus, the method has identified that line 1 of Figure 4 is not a part of a table as is the case with line 4 of Figure 1.

This exercise is repeated for each of the lines of text to be analysed from Figure 4 until a match or closest match has been identified for each of the lines. The table below shows another table of distance values obtained when comparing the sixth line of the text from Figure 4 with each of the lines of text from the training example text shown in Figure 1:

D(S1,T6)	1.890625
D(S2,T6)	2.453125
D(S3,T6)	2.4375
D(S4,T6)	3.835938
D(S5,T6)	2.4375
D(S6,T6)	3.835938
D(S7,T6)	2.4375
D(S8,T6)	3.835938
D(S9,T6)	2.648438
D(S10,T6)	2.296875
D(S11,T6)	1.34375
D(S12,T6)	2.09375
D(S13,T6)	3.382813
D(S14,T6)	3.390625
D(S15,T6)	2.382813

In this example, the smallest distance $D(S,T)$ is represented by line 11 of training example text S. Thus, the method concludes that the sixth line of text sample T is part of a table.

By identifying the location of the lines which are part of a table, the method can identify the transition between normal parts of text and parts of the text which comprise a table so as to locate the table within the block of text.

It is to be appreciated that the present invention is not limited to the particular special characters, separator characters or list of feature values described above. It is envisaged that other characteristics of a given line of text can be used as part of the process for training and recognising tables within text. Such further characteristics can be used instead of or in addition to the various characteristics described above. It should be appreciated that other pattern recognition algorithms which are readily available could be used instead of the nearest neighbour algorithm described above.

**CLAIMS:**

1. A method of recognising the location of a table within a block of text comprising the steps of: attributing a set of values to each line of a sample block of text, which set of values are determined at least by the characteristics of the line of text, the set of values for the sample block of text comprising training examples; indicating whether each line includes part of a table; attributing a set of values to each line of a block of text to be analysed, which set of values are determined at least by the characteristics of that line of text; applying a pattern recognition algorithm to each line of the block of text being analysed and lines from the training examples to identify whether the line of the block of text being analysed includes a part of a table.
2. A method according to Claim 1, wherein the step of indicating whether each line includes part of a table is a manual step.
3. A method according to Claim 1 or 2, wherein the set of values attributed to each line of a block of text comprise three sets of values: a first set of values which are determined by the characteristics of the previous line of text; a second set of values which are determined by the characteristics of the present line of text; and a third set of values which are determined by the characteristics of the following line of text.
4. A method according to Claim 3, wherein each set of values comprises a number of feature values each of which represents a characteristic of a given line of text.
5. A method according to Claim 4, wherein the feature values represent one or more of the following characteristics of a given line of text: whether or



not the line is a blank line having no text characters; the number of initial space characters in the line before the first text character; an indication of the first non-space character of the line; an indication of the last non-space character of the line; and indication whether the line consists entirely of one special character only; a number of segments in the line with two or more contiguous space characters; a number of segments in a line with three or more contiguous space characters; the number of segments in the line with two or more contiguous separator characters; the number of segments in the line with three or more contiguous separator characters.

6. A method according to Claim 4 or 5, wherein the feature value comprises a numerical value or a symbolic value represented by a special character.

7. A method according to any preceding claim, wherein the pattern recognition algorithm comprises a nearest neighbour algorithm.

8. A method according to any preceding claim, wherein the pattern recognition algorithm provides an indication of the line or lines from the sample block of text which are closest to a given line of text from the block of text being analysed.

9. A method according to Claim 8, wherein a given line is recognised as being part of a table or not if the line or lines from the sample block of text which are closest to the given line of text being analysed comprise a part of a table or not.

10. A method of producing training examples of sample text to be used with the method of any one of Claims 1 to 9, comprising the steps of: attributing a



set of values to each line of a sample block of text, which set of values are determined at least by the characteristics of the line of text, the set of values for the sample block of text comprising training examples; and indicating whether each line includes part of a table.