

**21st Meeting of the Wiesbaden Group on Business Registers
– International Roundtable on Business Survey Frames**

24-27 November 2008, OECD Paris, France

**Germany
Patrizia Moedinger
Federal Statistical Office
November 13th 2008**

Session 5 - Projects on Improvement for Business Register

Application of regular expressions in the German Business Register

Abstract

Regular expressions, often called patterns, are expressions that describe a set of strings. Regular expressions are written in a formal language that can be interpreted by a program that examines text and identifies parts that match the provided specification. Some Business Register (BR) variables, for example enterprise addresses, consist of text data. In data processing, text data requires special treatment. So, regular expressions are a powerful, flexible and efficient tool for identifying and editing text data (strings).

The paper gives two examples of using regular expressions for text data editing in the German Business Register. The first example describes how enterprise names can be used for automatic legal form coding. The second example deals with the possibilities and benefits of using regular expressions for data pre-processing as a preliminary for record linkage.

1. Improving legal form coding by using regular expressions

1.1 Background

In the German Business Register, the information concerning the legal form is obtained from administrative sources, mainly from VAT records. But not all administrative sources provide information on legal forms and the diverse sources use different not compatible legal form coding or different aggregation levels. Furthermore, the legal form coding provided by the administrative sources doesn't meet the requirements for other purposes like e. g. the coding of institutional sectors.

In Germany, enterprises (legal units) with certain legal forms are legally obliged to carry their legal form in the enterprise name. This applies to incorporated and non-incorporated firms, cooperatives

and merchants that are registered in the German Commercial Register. In these cases enterprise names can be used for legal form coding.

The use of enterprise names for legal form coding has two objectives. Primarily, the quality of the legal form coding shall be improved by using the enterprise name as a new source of coding beside administrative sources. Secondly, the enterprise name can be used to identify specific legal forms that are used for institutional sector coding.

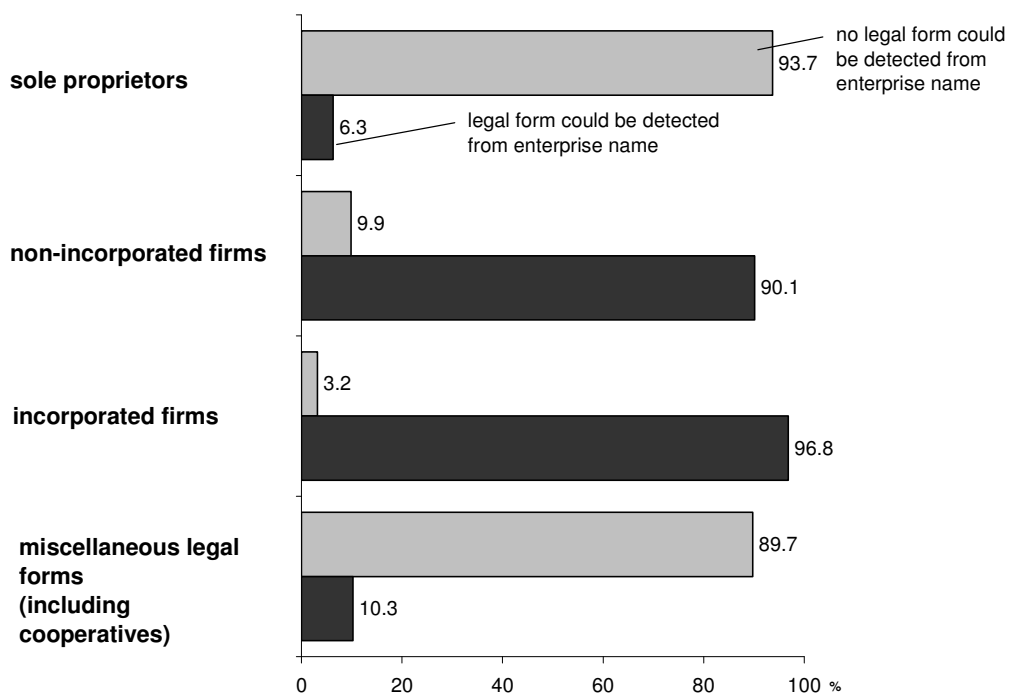
1.2. Definition of search patterns

The first step in defining search patterns for legal form coding is to use the nomenclature, abbreviation and notations from the tax authorities. The formal list of search patterns is completed with patterns that appear in the BR real data. Based on this list, the regular expressions for searching legal forms in enterprise names are constructed.

1.3. Evaluation of search patterns

To start with the evaluation of legal form coding with regular expressions, we look at the completeness of the automatic coding. As there is a legal obligation to carry information on certain legal forms in the enterprise name, we should observe a high level of found patterns for these legal forms.

Figure 1 Proportions of found legal forms in enterprise names by legal form



Despite the legal obligation for incorporated and non-incorporated firms to carry information on their legal form in their names, the program does not find legal forms in all incorporated and non-incorporated enterprise names. An explanation is that the search patterns of the program are incomplete. An alternative explanation is that the data is fragmentary and there is no legal form in the company's name.

The user that relies on the automatic legal form coding should know its degree of reliance. The program should detect a legal form in the name if there is one (true – positive). And the regular expression should not detect a legal form, if there is none (true – negative). Based on this, we can distinguish two error types: The name contains no legal form and the program finds one (false – positive or Type I error). Or there is a legal form in the name and the program doesn't identify it (false – negative or Type II error).

Table 1: Classification of Evaluation Outcome

		Enterprise name contains	
		legal form	no or wrong legal form
regular expression detects	legal form	True – positive	Type I Error (false – positive)
	no or wrong legal form	Type II Error (false – negative)	True – negative

To analyse the degree of reliance a 0.1 % random sample of the active enterprise population was drawn after automatic coding. From that sample the Type I error - all cases where legal forms were misleadingly found and Type II error - all cases where the legal forms were misleadingly not found, were counted.

Table 2: Evaluation of Type I and II Errors

		Enterprise name contains		
		legal form	no or wrong legal form	
regular expression detects	legal form	1,009	4	PPV (positive predictive value) = 1,009 / (1,009 + 4) = 99.6 %
	no or wrong legal form	26	2,961	NPV (negative predictive value) = 2,961 / (2,961 + 24) = 99.1 %
		Sensitivity = 1,009 / (1,009 + 26) = 97.5 %	Specificity = 2,961 / (4 + 2,961) = 99.8 %	N =4,000

In 97.5 % the program identifies legal forms the enterprise name, if there is one. In 99.8 % the program identifies no legal form in the enterprise name, if there is none. In 99.6 % the legal form is correctly predicted. In 99.1 % the program correctly predicts no legal form in the enterprise name. Even though the sample size is quite small, the results of the automatic legal form coding seem highly reliable.

2. Data pre-processing as a preliminary for record linkage

2.1. Background

The German BR is updated by using different administrative data sources. As there are no common unique identifiers available, the data from different sources are initially linked by names and addresses. A problem associated with data matching is that the administrations use different or none address standards. To give an example, the notation “BMW“ or “Bayerische Motorenwerke“ or “Bay. Motorenwerke“ is labelling the identical entity.

To illustrate the problem of non standardized notation in enterprise names and addresses within administrative sources and between administrative sources and the BR, we'll look at the matching by administrative identifiers. We consider a successful match where the data matches by administrative identifiers and where is no change in the postal code¹. For independent variable we look at the amount of differences between e.g. enterprise names, street names and town names. The difference between two strings is measured by the Levenshtein edit distance that is given by the minimum number of operations needed to transform one string into the other, where an operation is an insertion, deletion, or substitution of a single character. For easier comparison, the Levenshtein edit distance is divided by the maximum string length². With a logistic regression model the matching probabilities against string similarity can be estimated.

The results of the logistic regression estimations are shown in figure 2 and 3. The y-axis displays the predicted match–probabilities. The x-axis shows the Levenshtein edit distance divided by the maximum string length. Even though the data is coming from the same administrative source, the similarity of enterprise names and addresses is not a good indicator for a successful matching (Figure 2). If the data is coming from different sources, the disparity in enterprise names and addresses even for matched data will be huge (Figure 3).

For successful record linkage on enterprise names and addresses a high level of similarity between two strings should indicate identical units and a high level of disparity between two strings should indicate different units.

¹ This is a rough estimate of true matches because the unit behind the administrative identifier can change and there are normal changes in address and names.

² The use of the maximum string length tends to overestimate the string disparity in cases where the length of the two strings differs highly.

Figure 2: Matching probability against string similarity within an administrative source (Employment Agency) (Model: Logistic regression)

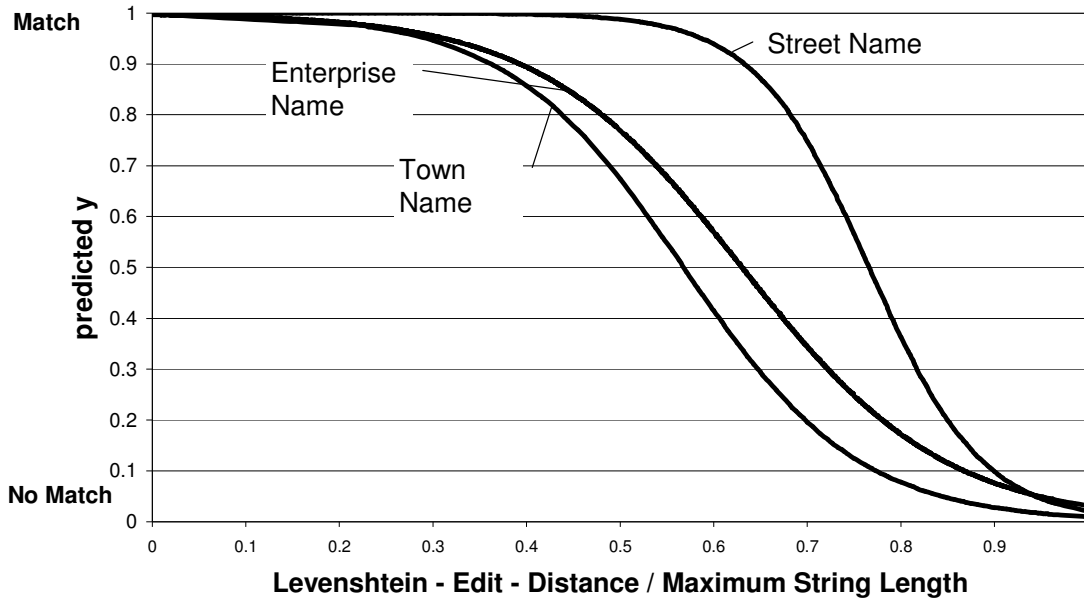
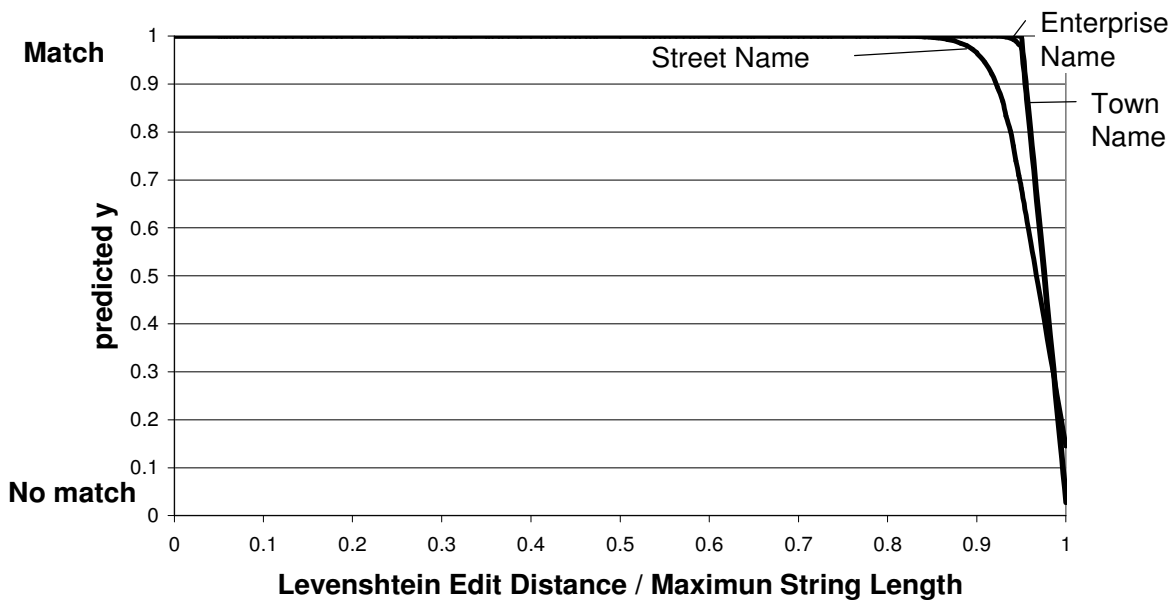


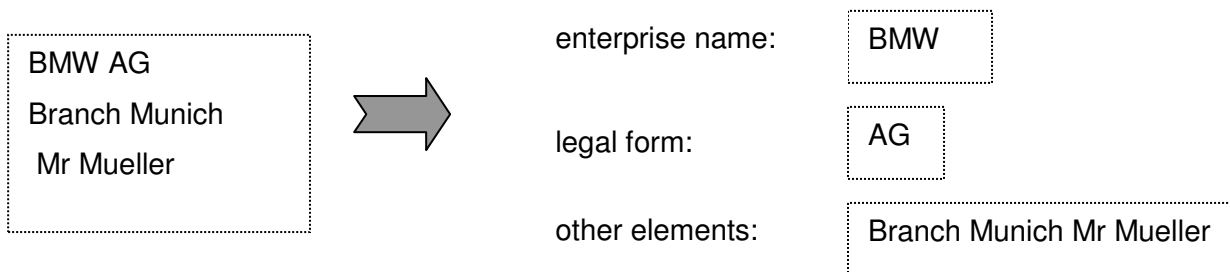
Figure 3: Matching probability against string similarity between an administrative source (Employment Agency) and BR (Model: Logistic regression)



2.2. Pre-processing administrative data for record linkage

Before matching by enterprise name and addresses the input administrative data and the BR should be prepared. This pre-processing of input data standardizes the variables that are used for record linkage. For successful matching all pre-processing activities should improve the link between string similarity and real match. So that a high level of similarity between two strings indicates a match and a high level of disparity means no match.

For a start the non-standardized variables of the administrative source are converted into specific variables for string matching. To give an example, the enterprise name is split into the three variables; namely exclusive enterprise name, legal form notation and other elements like branch or contract person.



This means that the variables have similar content for later comparison. To control different notations or misspellings, the comparison strings are simplified. Regular expressions can be used in both steps: regular expressions search for patterns, fill in specific variables and perform data editing for simplifying the strings for later comparison. With in-depth pre-processing the correlation among string similarity and match should improve.

3.3 Evaluation

A potential approach for evaluation is looking at the results of the automatic matching before any manual work is done. The basic matching program uses the postal code for grouping the data before comparing string similarities and makes the robust assumption that a high string similarity indicates a match.

As pointed out before the matching program should link units if they are truly matched (true – positive) and should not match units if they are truly different (true – negative). Based on this, we can distinguish two error types: The program matches units that are truly different (false – positive or Type I error) or the units belong together and the program doesn't identify them (false – negative or Type II error).

Table 3: Evaluation of automatic matching result BR with external reporting units

		True		
		Match	No Match	
Program	Matches	1,807	957	PPV (positive predictive value) = 1,807 / (1,807 + 957) = 65.4 %
	doesn't match	3,515	1,156	NPV (negative predictive value) = 1,156 / (3,515 + 1,156) = 24.7 %
		Sensitivity = 1,807 / (1,807 + 3,515) = 34 %	Specificity = 1,156 / (1,156 + 957) = 54.7%	N =7,435

For a start Table 3 displays the results of the automatic matching between the BR and external reporting units. The results refer to non pre-processed data. The correlation among string similarity and match is quite weak. In 34 % the program matched the BR units with the external reporting units correctly. In 54.7 % the program identifies no match, if there is none. In 65.4 % the match is correctly predicted. In 24.7 % the program correctly predicted that there is no match.

To improve the matching results the data on enterprise names and addresses used for record linkage have to be pre-processed. The variables used for matching records should indicate a high level of similarity between the strings if there is a match and should report a high level of disparity if they are different units. A powerful tool for doing so is using regular expressions.