Multiclass Text Categorization for Automated Survey Coding

Daniela Giorgetti
Istituto di Linguistica Computazionale
Consiglio Nazionale delle Ricerche
56124 Pisa, Italy
daniela.giorgetti@ilc.cnr.it

Fabrizio Sebastiani
Istituto di Scienza e Tecnologie dell'Informazione
Consiglio Nazionale delle Ricerche
56124 Pisa, Italy
fabrizio@iei.pi.cnr.it

ABSTRACT

Survey coding is the task of assigning a symbolic code from a predefined set of such codes to the answer given in response to an open-ended question in a questionnaire (aka *survey*). We formulate the problem of automated survey coding as a *text categorization* problem, i.e. as the problem of learning, by means of supervised machine learning techniques, a model of the association between answers and codes from a training set of pre-coded answers, and applying the resulting model to the classification of new answers. In this paper we experiment with two different learning techniques, one based on naïve Bayesian classification and the other one based on multiclass support vector machines, and test the resulting framework on a corpus of social surveys. The results we have obtained significantly outperform the results achieved by previous automated survey coding approaches.

Keywords

Open-ended survey coding, multiclass text categorization

1. INTRODUCTION

Survey coding is the task of assigning a symbolic code from a predefined set of such codes to a textual expression representing the answer that a person has given in response to an open-ended question of a survey. By *open-ended* we mean a question that requires or allows an answer consisting of free text; open-ended questions are the opposite of *multiple-choice* questions, which instead require to select the answer from a predefined set. Survey coding has several applications, especially in the social sciences, ranging from the simple classification of respondents to the extraction of statistics on political opinions, health and lifestyle habits, customer satisfaction, brand fidelity, and patient satisfaction.

As an example, in 1996 interviewers asked the following question (among many) to a carefully chosen sample of 1370 subjects, in the framework of the General Social Survey [4] carried out by the US National Opinion Research Center (NORC)[1]:

> `Within the past month, think about the last time you felt really angry, irritated or annoyed. Could you describe in a couple of sentences what made you feel that way - what the situation was?`

Professional coders were then asked to classify the answers in exactly one among the following categories, each consisting of a code label and a short explicatory caption:

ANGRYWRK:	`Situation involved work`
ANGRYFAM:	`Situation involved family`
ANGRYGVT:	`Situation involved government or government officials`
WRK&FAM:	`Situation involved both work and family`
WRK&GVT:	`Situation involved both work and government`
FAM&GVT:	`Situation involved both family and government`
OTHER:	`Situation did not fit the above categories`

Answers included for example:

> `trying to teach my son something and he was being stubborn and wouldn't listen to me i got angry at him`

which coders classified under the ANGRYFAM header.

Survey coding is a difficult task, since the code that should be attributed to a respondent based on the answer she has given is a matter of subjective judgment, and thus requires expertise. For instance, different coders, especially if little trained, might have different opinions as whether the answer

> `when people in authorities arent treating people right`

should be classified under ANGRYGVT, or ANGRYWRK, or WRK&GVT, or even under OTHER. Given the difficulty of the task, it is thus unsurprising that it has traditionally been performed manually, by professional coders.

Some attempts have been made in the past at automating the survey coding task. Most of them have exploited simple techniques from the tradition of text retrieval, for matching the answer and textual descriptions of the meanings of the candidate codes [13]. In this paper we take a radically new stand, and formulate the problem of automated survey coding as a *(multiclass) text categorization*

[1] `http://www.norc.uchicago.edu/`

SAC 2003, Melbourne, Florida USA

problem, i.e. as the problem of learning, by means of supervised learning techniques, a model of the association between answers and codes from a training set of pre-coded answers, and applying the resulting model to the classification of new answers into exactly one of the predefined code categories. In this paper we experiment with two different supervised learning techniques (one based on naïve Bayesian classification [8, 9], and another based on multiclass support vector machines [6]) and test the resulting framework on a corpus of social surveys conducted by NORC. The results we obtain significantly outperform the results achieved by previous automated survey coding approaches.

This paper is structured as follows. Section 2 introduces survey coding, and reviews related work attempting to automate this task. In Section 3 we describe how the survey coding task can be framed as a multiclass text categorization problem. Section 4 illustrates the experiments we have performed in the application of naïve Bayesian classification and support vector machines to the problem of coding a corpus of social surveys collected by NORC. Section 5 concludes, commenting our results and discussing possible avenues for further research.

2. SURVEY CODING AND ITS AUTOMATION

The survey coding process is both slow and expensive, since a lot of manual effort by different professional figures is involved. For example, NORC interviewers take handwritten notes of the answers returned during the interview, and typists produce a typewritten text from these notes at a later stage; this text is then analyzed by professional coders, who perform the final coding task. Yet another drawback is that the process is likely to produce faulty encodings, as there are several potential sources of error: interviewers may misunderstand the answers or misrepresent them by their notes, typists may misread or misunderstand the handwritten notes or introduce further typing errors, and coders may misinterpret the meaning either of the answers or of the codes. Our automated approach currently deals only with the coding phase on transcript data; an even better approach would be to code "first-hand data", i.e. data transcribed directly from speech (see Section 5 for a discussion).

Given that text analysis for the social sciences is an important problem, several software packages that address it have been developed. However, they are not usually tailored for the specific task of survey analysis, and the solutions that they provide for the survey coding task are fairly unsatisfactory. Most of these software packages (see [1] for a review) mostly concentrate on helping coders in coding their data *manually*, and in visualizing them in several convenient ways. A few of these packages instead do perform automatic coding, by relying mostly on specialized dictionaries. This means that text fragments are automatically assigned to a specific category if and only if they contain words matching those in the dictionary relevant to the category. One of the disadvantages of this approach is that dictionaries have to be created *before* the coding process begins, i.e. when data is still totally unknown; this approach is thus extremely static. The second drawback is that specialized dictionaries need to be developed, one for each category of interest; this requires the intervention of expert personnel, who is then responsible for deciding which words, if present (either alone or in combination) or absent in an answer, should trigger the attribution of the code to the answer.

The scientific literature on automating survey coding is very scarce. The approach that is closest in spirit to ours is probably the dictionary-based approach as it is described in Viechnicki's work [13]. In this paper responses to questions from a NORC survey are classified by means of a set of codes pre-defined by NORC social scientists. Viechnicki proposes two alternative approaches. In the former, words that characterize a given category are combined by means of Boolean operators, and the answer is classified under the category whose Boolean description it matches. The latter method is instead based on computing the similarity between two weighted vectors of words extracted from the answer and from a textual explicatory caption of the code, and choosing the code with the highest similarity score.

Our approach has several advantages with respect to the dictionary-based approach. First, in our learning-based approach the manual effort is directed towards the manual coding of a small training set of answers, and not towards the creation of specialized dictionaries. This is advantageous, as it is easier to characterize a concept extensionally selecting instances of it, e.g., manually assigning codes to a set of documents, than intensionally, describing the concept in words or describing a procedure for recognizing its instances, e.g., building and tuning a dictionary of words that trigger the attribution of a code. Second, our approach is solidly grounded in machine learning theory, and it can leverage on a wealth of results and techniques developed within text categorization. Of course, our approach is mostly useful for medium- to large-sized surveys, as in the learning phase we need a hand-coded set of answers to train the inductive learner. This means that if a survey is somewhat limited in the number of surveyed people, hand-coding the training set may coincide with hand-coding the entire set.

3. AUTOMATED SURVEY CODING BY TEXT CATEGORIZATION

Text categorization (TC) is the task of automatically building, by means of machine learning (ML) techniques, *automatic text classifiers*, i.e. programs capable of labelling natural language texts from a domain $\mathcal{D}$ with thematic categories from a predefined set $\mathcal{C} = \{c_1, \ldots, c_{|\mathcal{C}|}\}$. The task where each document in $\mathcal{D}$ must be tagged with exactly one category from $\mathcal{C}$ is called *multiclass* TC [5]. Since survey coding is typically a multiclass TC task, in the rest of the paper we will always refer to the multiclass case. In multiclass TC effectiveness is measured in terms of *accuracy*, defined as the ratio between the number of correct classification decisions and the total number of classification decisions.

The construction of an automatic text classifier relies on the existence of a *labelled corpus* $\Omega = \{d_1, \ldots, d_{|\Omega|}\}$ of documents preclassified under $\mathcal{C}$. A general inductive process (called the *learner*) automatically builds a classifier for $\mathcal{C}$ by learning the characteristics of $\mathcal{C}$ from a *training set* $Tr = \{d_1, \ldots, d_{|Tr|}\}$ of documents. Once a classifier has been built, its effectiveness (i.e. its capability to take the right categorization decisions) may be tested by applying it to the *test set* $Te = \Omega - Tr$, and checking the accuracy of the results. In our survey coding context, the set of all answers to a given question q play the role of $\mathcal{D}$, and the set

of all possible codes that may be attributed to an answer to question q play the role of $\mathcal{C}$ (different questions correspond thus to different TC tasks).

The input to the learners (and to the classifiers, once they have been built), consists of an answer d_j represented as a vector of term *weights* $\vec{d_j} = \langle w_{1j}, \ldots, w_{|\mathcal{T}|j} \rangle$. Here, $\mathcal{T}$ is the *dictionary*, i.e. the set of words that occur at least once in the training set, and $0 \leq w_{kj} \leq 1$ quantifies the importance of t_k in characterizing the semantics of d_j.

Several methods have been proposed in the text categorization literature for learning a text classifier from training data (see [11] for a review). In this work we have run a series of experiments with two different classifier-learning methods. The first learner we use is a probabilistic naïve Bayesian learner, as implemented in the RAINBOW package[2]. Probabilistic text classification methods assume that the data was generated by a parametric model, and use the training data to estimate the parameters of this model. Bayes' theorem allows to estimate from this model the probability that a given category has generated the document to be classified; classification thus consists in selecting the category with the highest probability. There are two well-known variants of this method, the multi-variate Bernoulli method and the multinomial method. In this paper we chose the latter, since in comparative text classification experiments it has performed better than the former [9].

The second learning method is a multiclass support vector machine (SVM) learner as embodied in the BSVM package[3]. SVMs attempt to learn a hyperplane in $|\mathcal{T}|$-dimensional space that separates the "positive training examples" of category c_i from the negative ones with the maximum possible margin, i.e. such that the minimal distance between the hyperplane and a training example is maximum; results in computational learning theory indicate that this tends to minimize the generalization error, i.e. the error of the resulting classifier on yet unseen examples. SVMs were usually conceived for binary classification problems (where $|\mathcal{C}| = 2$), and only recently they have been adapted to multiclass classification.

With respect to effectiveness, the text categorization literature has shown that naïve Bayesian approaches are, relatively to other learning methods, no more than average performers (see e.g. [11, Section 7]). On the contrary, the same literature has shown that support vector machines are (together with "boosting"-based classifier committees) the unsurpassed top performers. The reason why we experiment with RAINBOW is that we want to show that a text categorization approach to survey coding is much more effective than the dictionary based approach *regardless of the specific learning method adopted*, i.e., even with an average-performing learning method our text categorization approach to survey coding can largely outperform the dictionary-based method. Instead, the reason we experiment with BSVM is that we want to show what level of effectiveness this approach can achieve, once instantiated with a top-performing learning algorithm.

We have used a binary representation as input to RAINBOW, and a non-binary one as input to BSVM. This is due to the fact that the probabilistic models upon which RAINBOW is based require binary inputs, while this is not the case for SVMs. In the binary representation, w_{kj} represents just presence or absence of term t_k in answer d_j. Our non-binary representation is instead the $tfidf$ function in its standard "ltc" variant [10], i.e.

$$tfidf(t_k, d_j) = tf(t_k, d_j) \cdot \log \frac{|Tr|}{\#_{Tr}(t_k)} \quad (1)$$

where $\#_{Tr}(t_k)$ denotes the number of answers in the training set Tr in which t_k occurs at least once and

$$tf(t_k, d_j) = \begin{cases} 1 + \log \#(t_k, d_j) & \text{if } \#(t_k, d_j) > 0 \\ 0 & \text{otherwise} \end{cases}$$

where $\#(t_k, d_j)$ denotes the number of times t_k occurs in answer d_j. Weights obtained by Equation 1 are normalized by cosine normalization, yielding

$$w_{kj} = \frac{tfidf(t_k, d_j)}{\sqrt{\sum_{s=1}^{|\mathcal{T}|} tfidf(t_s, d_j)^2}} \quad (2)$$

In all the experiments discussed in this paper, stop words, punctuation, and numbers, have been removed, and all letters have been converted to lowercase. No feature selection (see e.g. [11, Section 5.1]) has been performed. The reason is that, as shown in extensive experiments by Brank et al. [2], the effectiveness of SVMs is usually worsened by feature selection, irrespectively of the feature selection algorithm used and of the chosen reduction factor (this is also independently confirmed by the results of [12]), and the effectiveness of naïve Bayesian methods does not show systematic patterns of improvement either.

4. EXPERIMENTS

As already pointed out, our experiments have been carried out on data from NORC's General Social Survey. This survey, which is ongoing since 1972, aims at investigating how people assess their physical and mental health, the balancing of security and civil liberties, external and internal security threats, intergroup relations and cultural pluralism, religious congregations, etc. We deal with three datasets (see Table 1) from the NORC General Social Survey administered in 1996. Each of these datasets (here nicknamed *angry_at*, *angry_why*, and *brkdhlp*) consists of a set of answers to a given question, plus their associated category codes manually chosen by NORC's professional coders from a predefined set of category codes. The task consists in choosing exactly one category code for each answer[4].

We have chosen these three datasets because they are the same datasets used in [13], which means that we will be able to obtain a direct comparison between the effectiveness of Viechnicki's method (which is representative of the dictionary-based approach to survey coding) and the effectiveness of our supervised learning approach. Note that all

[2]RAINBOW can be downloaded from `http://www.cs.cmu.edu/~mccallum/rainbow`.

[3]BSVM is available from `http://www.csie.ntu.edu.tw/~cjlin/bsvm/`.

[4]The *angry_at* and *angry_why* datasets actually involve the same question, which deals with the description of a situation that caused anger to the respondent; each answer was classified according to *two* different sets of codes, one concerning the object of anger, the other concerning the cause of anger (actually, *angry_why* contains only a subset of the answers contained in *angry_at*, in the sense that NORC coders classified some of the answers only according to the *angry_at* set of codes). The *brkdhlp* dataset (called *breakdown* in [13]) consists of answers to the question as to what source of help was used to deal with a nervous breakdown.

Dataset	Category	# of instances
angry_at	ANGRYFAM	275
	ANGRYWRK	345
	ANGRYGVT	74
	WRK&GVT	8
	WRK&FAM	27
	FAM&GVT	16
	OTHER	625
	total	1370
angry_why	SELF	29
	PREVENTED	36
	CRITICAL	88
	DEMANDING	60
	EXPECT	196
	OTHER	51
	total	460
brkdhlp	FAMILY	57
	FRIEND	33
	GROUP	2
	CLERGY	55
	PSYCHIATRIST	56
	AGENCY	16
	OTHER	148
	total	367

Table 1: Characteristics of the three datasets used in our experiments.

three datasets include a class OTHER. This consideration alone indicates that these datasets are not "easy", since

- the classifier cannot simply "take a guess" by picking "the least inappropriate" category, since if all choices are sufficiently inappropriate, the category OTHER applies;
- the category OTHER will typically be a very hard category to work with, since it will not be characterized by a specific terminology, as is instead the case with categories that are strongly characterized in a topical sense. The presence of a category OTHER in the category set always tends to deteriorate the global performance of any text classifier.

For each dataset, the main steps we went through to run our experiments are the following:

1. preprocess the data in order to obtain a data format compatible with the learners (this had to be repeated once for RAINBOW and once for BSVM, since they require different input formats);
2. partition the set of answers in each dataset in four random disjoint subsets of equal size;
3. run the learner to generate a classifier, using one of the four subsets as the test set and the other three as the training set;
4. run the classifier to classify the data in each test set of each dataset and evaluate the results.

In order to achieve better statistical significance, in all experiments steps 3 and 4 were repeated four times, for all four possible choices of the test set. The results we report are thus averaged across four different experiments.

We have computed accuracy on the three datasets both with RAINBOW and BSVM; the results are reported in Table 2, where they are compared with the accuracy obtained in [13] on the same datasets.

The first observation we can make is that the supervised learning approach to survey coding significantly outperforms

	Dictionary-Based [13]		Supervised Learning	
	Vector	Boolean	RAINBOW	BSVM
angry_at	0.451	0.465	**0.714 (+54%)**	0.693 (+49%)
angry_why	0.211	0.272	0.389 (+43%)	**0.397 (+46%)**
brkdhlp	0.646	**0.747**	0.653 (-13%)	0.643 (-14%)
Average	0.436	0.495	**0.585 (+18%)**	0.578 (+17%)
Std. Dev.	0.218	0.239	0.173 (-28%)	**0.158 (-34%)**

Table 2: Comparative accuracy results obtained on the *angry_at*, *angry_why* and *brkdhlp* datasets using a Boolean and a vector-based method and using a naïve Bayesian and a multiclass SVM TC methods. The percentile improvements in accuracy and average accuracy, and the percentile reductions in standard deviation, are reported with respect to the Boolean method, the best dictionary-based method in [13]. Boldface indicates the best performance on the dataset.

the dictionary-based approach: the improvements with respect to the best-performing method reported in [13] are significant, a +18% on average for RAINBOW and a +17% for BSVM. The improvement is especially significant on the "non-obvious" datasets: for instance, *angry_why* appears to be a hard to characterize dataset, as shown from the poor performance of the two dictionary-based methods, and on this dataset the supervised learning methods improve up to +88% with respect to them. On the contrary, the *brkdhlp* dataset seems easy to tackle by simple Boolean rules, as shown by the .747 accuracy figure of the Boolean method; in this case the supervised learning methods underperform the Boolean method by up to 14%.

Moreover, the supervised learning approach delivers a more stable performance across the three datasets, since the reductions in standard deviation with respect to the same best-performing method are very significant, a -28% for RAINBOW and a -34% for BSVM.

As anticipated in Section 3, the fact that improvements of this order of magnitude are obtained even with a method, such as the naïve Bayesian technique implemented in RAINBOW, which is known as an average performer in the text categorization literature, bears witness to the superiority of the supervised learning approach to survey coding.

The fact that multiclass SVMs, known top-performers in the machine learning literature (see e.g. [3]), underperform RAINBOW, even though by a very small margin, is more surprising, and might be due to our using BSVM with a naive parameter setting. We plan to explore the BSVM parameter space more thoroughly in our next experiments. A further possible explanation is that, as widely believed in the machine learning community, the solution to the multiclass SVM problem implemented in BSVM is suboptimal. A more satisfactory solution to this problem could be the one proposed by Crammer and Singer [3]; we plan to experiment with their system as soon as it is released.

5. CONCLUSION

We have shown that automatic coding of responses to open-ended survey questions may be posed as a multiclass text categorization problem, and that text categorization techniques based on supervised learning significantly outperform the dictionary-based techniques that have been up

to now the dominant approach to automated survey coding. Another advantage of the supervised learning approach with respect to the dictionary-based approach, which requires that the text classifiers be handcrafted (by a knowledge engineer and a social scientist working together), is that the classifiers can be generated automatically from the training data, with substantive savings in terms of expert manpower.

The effectiveness levels that text categorization techniques have achieved in our experiments are far from being perfect, and also from being completely satisfactory. Although the results obtained in our research are promising, we think that more research is needed for the automatic approach to survey coding to clearly supersede the manual approach. There are several avenues for further research. One of these, which we are currently working at, is simply to run experiments on more survey data, in order to obtain results which are statistically more reliable. Another possible line of research is to experiment with more satisfactory multiclass TC learners, in order to improve upon the results of RAINBOW and BSVM.

In the future, we plan to combine automated survey coding by text categorization with speech recognition, in order to allow the survey coding task to proceed directly from the audio recording of the interview, since we believe that survey coding may be performed with much better effectiveness only by using better quality input, i.e. more faithful representations of the answers. Proceeding directly from the audio recording can eliminate the sources of noise mentioned in Section 2 (i.e. the noise possibly introduced by interviewers and typists), and also makes for greater savings in term of manpower, which means that the researchers who design the survey could afford having more open-ended questions and less multiple-choice ones. However, such a process would entail the need to apply text categorization techniques to noisy text, since speech recognition software performs imperfectly, especially in dealing with natural speech in possibly noisy environments. To this respect, we think that there are reasons for optimism, since research in text categorization of noisy text [7] has already shown that, by employing noisy texts also in the training phase (i.e. texts affected by the same source of noise that is also at work in the test documents), effectiveness levels comparable to those obtainable in the case of standard text can be achieved. Although the source of noise tackled in [7] was different (i.e. noise resulting from optical character recognition), similar effectiveness patterns might result also in the case of noise introduced by speech recognition.

Acknowledgements

The open-ended text used in this work was collected in the General Social Surveys of the National Opinion Research Center (NORC), University of Chicago, and supplied by NORC to the authors. We are grateful to Tom Smith and Jennifer Berktold for providing these texts and for assisting us in their interpretation. We are also grateful to Chih-Wei Hsu, Chih-Jen Lin and Andrew McCallum for making the BSVM and RAINBOW packages available, to Henri Avancini for helping with preprocessing issues, to Peter Viechnicki for clarifying several points of his experiments, and to David Lewis for suggesting pointers to the automated survey coding literature.

6. REFERENCES

[1] M. Alexa and C. Züll. Text analysis software: Commonalities, differences and limitations. The results of a review. *Quality and Quantity*, 34:299–321, 2000.

[2] J. Brank, M. Grobelnik, N. Milić-Frayling, and D. Mladenić. Interaction of feature selection methods and linear classification models. In *Proceedings of the ICML-02 Workshop on Text Learning*, Sydney, AU, 2002.

[3] K. Crammer and Y. Singer. On the algorithmic implementation of multiclass kernel-based vector machines. *Journal of Machine Learning Research*, 2:265–292, 2001.

[4] J. A. Davis and T. Smith. *General Social Surveys, 1972-1996: Cumulative Codebook*. National Opinion Research Center, Chicago, US, 1996.

[5] C.-W. Hsu and C.-J. Lin. A comparison of methods for multi-class support vector machines. *IEEE Transactions on Neural Networks*, 13(2):415–425, 2002.

[6] C.-W. Hsu and C.-J. Lin. A simple decomposition method for support vector machines. *Machine Learning*, 46:291–314, 2002.

[7] D. J. Ittner, D. D. Lewis, and D. D. Ahn. Text categorization of low quality images. In *Proceedings of SDAIR-95, 4th Annual Symposium on Document Analysis and Information Retrieval*, pages 301–315, Las Vegas, US, 1995.

[8] D. D. Lewis. Naive (Bayes) at forty: The independence assumption in information retrieval. In C. Nédellec and C. Rouveirol, editors, *Proceedings of ECML-98, 10th European Conference on Machine Learning*, pages 4–15, Chemnitz, DE, 1998. Springer Verlag, Heidelberg, DE. Published in the "Lecture Notes in Computer Science" series, number 1398.

[9] A. K. McCallum and K. Nigam. A comparison of event models for naive Bayes text classification. In *Proceedings of the 1st AAAI Workshop on Learning for Text Categorization*, pages 41–48, Madison, US, 1998.

[10] G. Salton and C. Buckley. Term-weighting approaches in automatic text retrieval. *Information Processing and Management*, 24(5):513–523, 1988.

[11] F. Sebastiani. Machine learning in automated text categorization. *ACM Computing Surveys*, 34(1):1–47, 2002.

[12] H. Taira and M. Haruno. Feature selection in SVM text categorization. In *Proceedings of AAAI-99, 16th Conference of the American Association for Artificial Intelligence*, pages 480–486, Orlando, US, 1999. AAAI Press, Menlo Park, US.

[13] P. Viechnicki. A performance evaluation of automatic survey classifiers. In V. Honavar and G. Slutzki, editors, *Proceedings of ICGI-98, 4th International Colloquium on Grammatical Inference*, pages 244–256, Ames, US, 1998. Springer Verlag, Heidelberg, DE. Published in the "Lecture Notes in Computer Science" series, number 1433.