Document Difficulty Framework for Semi-automatic Text Classification

Miguel Martinez-Alvarez[1], Alejandro Bellogin[2], and Thomas Roelleke[1]

[1] Queen Mary, University of London
{miguel,thor}@eecs.qmul.ac.uk
[2] Centrum Wiskunde & Informatica (CWI)
alejandro.bellogin@cwi.nl

Abstract. Text Classification systems are able to deal with large datasets, spending less time and human cost compared with manual classification. This is achieved, however, in expense of loss in quality. Semi-Automatic Text Classification (SATC) aims to achieve high quality with minimum human effort by ranking the documents according to their estimated certainty of being correctly classified. This paper introduces the Document Difficulty Framework (DDF), a unification of different strategies to estimate the document certainty, and its application to SATC. DDF exploits the scores and thresholds computed by any given classifier. Different metrics are obtained by changing the parameters of the three levels the framework is lied upon: how to measure the confidence for each document-class (evidence), which classes to observe (class) and how to aggregate this knowledge (aggregation). Experiments show that DDF metrics consistently achieve high error reduction with large portions of the collection being automatically classified. Furthermore, DDF outperforms all the reported SATC methods in the literature.

1 Introduction and Motivation

Automatic Text Classification (TC) provide much faster and cheaper classification than human experts. However, even though there have been large improvements in the last decades, human experts achieve higher quality. Since the introduction of automatic classifiers, two alternative options can be applied. Firstly, a full-automatic classification system is applied, where every document is classified according to the decisions made by the classifier. Secondly, a completely manual classification is performed, where human experts classify each document. The main drawbacks of the latter option are its huge cost and potential unfeasibility for large collections. On the other hand, the quality achieved by the manual approach will be higher. Full automatic TC is preferred if large datasets are used (i.e. webpage classification) or when lower quality is not as important as the human effort required. On the other hand, manual classification is the best option for systems that require high-quality and have medium size such as law or medical data. In general, the time saved by using an automatic system is leveraged with the possible quality loss with respect to the manual classification.

This research focuses on an intermediate solution using Semi-Automatic Text Classification (SATC) [8,1]. The main goal is to achieve high quality with minimum human

L. Bellatreche and M.K. Mohania (Eds.): DaWaK 2013, LNCS 8057, pp. 110–121, 2013.

effort or, more specifically, to use human experts only for the documents that the automatic system is more likely to misclassify. Therefore, maximising the quality, while minimising the cost. Given a set of documents to be labelled and a specific classifier, a SATC algorithm needs to rank the documents according to the likelihood of their classification decisions to be correct. The ranking allows experts to inspect documents iteratively, starting with the most uncertain ones, until a specific point, where the rest of the documents are automatically assigned. In addition, this strategy can be applied with variable resources over time (i.e. less human experts could be available).

This paper introduces the **Document Difficulty Framework (DDF)**, a family of document certainty algorithms, and its application to SATC. DDF exploits the document-class confidence scores computed by a classifier and the class thresholds given by any class-based thresholding strategy to calculate the certainty of each document. This implies that the class scores for all documents have to be computed. The framework defines an array of different metrics, depending on three different dimensions: how the document-class evidence is computed (evidence), which classes will be considered (class), and how to aggregate a document-based certainty (aggregation).

The remainder of this paper is organised as follows: Section 2 presents the background and related research. Section 3 introduces DDF, and analyses its different variations. Experiments are explained and analysed in Sections 4. Finally, Section 5 concludes the paper and presents the future work.

2 Background and Related Research

2.1 Multi-Label Text Classification

In multi-label TC (ML-TC), each document can belong to multiple classes. As a result, a classification process assigns a boolean value to each pair $(d_j, c_i) \in D \times C$, where D is a set of documents and C is a set of predefined categories [9]. To achieve this goal, most classifiers use a two-step procedure. Firstly, the classifier produces a score for each pair (d_j, c_i), and then a thresholding strategy decides, for each of the scores, if that value implies that the document belongs to the class (T) or not (F). Given a classifier σ and a threshold function δ, both processes can be mathematically denoted as: $\sigma \in \Sigma : D \times C \rightarrow [0, \infty]$, $\delta : \Sigma \times D \times C \rightarrow \{T, F\}$.

There are different types of thresholding strategies, based on documents or topics. For instance, using RCut [12], the R classes with higher score for a specific document are selected. On the other hand, SCut computes the threshold per class which maximises its quality (i.e. measured using F_1) [12]. If the number of training examples for a category is small, SCut can compute a very high or very low threshold from a global point of view. To address these drawbacks, SCutFBR.1 and SCutFBR.0 were introduced [12]. These metrics modify the behaviour of SCut in the case that the quality obtained for a class is lower than a value (*fbr*). The former strategy uses the highest ranked document as threshold value, while SCutFBR.0 assigns an infinite value. As a result, SCutFBR.0 does not assign any document in any class with lower quality than fbr.

2.2 Semi Automatic Text Classification (SATC)

SATC assumes that neither manual, nor full automatic classification is the optimum solution. This situation appears when full automatic classification achieves lower than required quality, and a full manual classification is either too expensive or unfeasible due to lack of resources. The foundation of SATC is that if we are able to separate the documents with high probability to be correctly classified, and the ones that are probably wrong, the latter can be inspected by human experts while the former will be automatically classified. As a result, the resources (the human experts) are optimised, while the quality remains high. To solve this task, SATC methods rank the documents to be classified according to their uncertainty. SATC assumes that the documents with higher certainty are probably better classified, whereas the documents with higher uncertainty are incorrectly classified. Therefore, the quality is maximised if the human annotators inspect the documents starting from the ones with higher uncertainty. The possibility of combining human and automatic classification has been suggested before [13,9,6]. However, only two approaches have been proposed: Document Difficulty and Utility-Theoretic Ranking. Both of them have been proven to be well-suited for SATC. Nonetheless, their performance have never been compared in the literature.

Document Difficulty [8] uses the classification scores and thresholds as evidence to compute the document certainty, where the labels with greater certainty are those with larger relative difference with respect to their threshold. The aggregation of label confidences is performed by averaging the confidences for those classes the document will be labelled in. The reason for the name similarity is that DDF extends and generalises the principles we explained in [8], mainly exploiting the classification scores and the threshold values within a classifier-independent framework. However, while only one method was proposed in that work, DDF represents a family of certainty metrics, where the previous metric appears naturally as an special case. In addition, the evaluation is also different. Our previous work evaluated the quality of the subset of automatically classified documents, instead of the whole collection. This research analyses the quality of the full test set, including both manual and automatic classified documents subsets.

The Utility-Theoretic Ranking (UT) method [1] optimises the global quality of the system, exploiting the potential benefit of manually inspecting each document, using the confidence scores of a classifier, and the gain in terms of quality that could be achieved, if that label is actually correct. The main conceptual difference with our approach is that UT exploits the collection information, whereas DDF focuses on each document independently. Furthermore, DDF exploits threshold information, and class filtering for the aggregated document certainty.

Similar to SATC, Active Learning (AL) ranks documents according to their benefit in the learning process, selecting which unlabelled documents should be manually labelled and being included as training examples. However, SATC focuses on the classification step, while AL operates in the training phase, selecting the documents from which the classifier can learn the most. Extensive research has been done related to single-label AL [7,10]. However, very limited research has tried to address the same problem in a multi-label environment [4,11].

2.3 Semi Automatic Text Classification Evaluation

SATC is evaluated using traditional classification quality measures such as micro-averaged-F$_1$, once the human and the automatic decisions have been combined. This approach, introduced by Berardi *et al.* [1], provides quality values for different proportions of the collection being automatically classified, where the most uncertain documents are manually classified. In addition, the goal of SATC is not only to compute the quality but to analyse how it varies depending on the number or documents considered. Therefore, quality variations with respect to the full automatic quality with the same classifier are also computed. The main issue is that the relative quality increase is fully dependent on the base quality, when all the documents are automatically classified. For instance, in some cases, a 100% quality increase is impossible (i.e. full automatic classification achieving 95% quality), while more than 100% is possible for others, making a comparison over different classifiers impossible. Berardi *et al.* [1] addressed these challenges and introduced two alternatives based on the error reduction with respect to the full automatic system, instead of its quality increase. Error Reduction at rank (ER) measures the error reduction with a specific number of documents being automatically classified, where $E_p(n)$ models the error (defined as 1-quality) achieved by a classifier p with n documents being manually classified,

$$\mathrm{ER}_p(n) = \frac{\mathrm{E}_p(0) - \mathrm{E}_p(n)}{\mathrm{E}_p(0)} \tag{1}$$

Normalised Error Reduction at rank (NER) subtracts the error reduction at rank n achieved by a random ranker ($\frac{n}{|Te|}$, where $|Te|$ is the size of the documents to be classified) from ER in order to obtain more meaningful quality values,

$$\mathrm{NER}_p(n) = \mathrm{ER}_p(n) - \frac{n}{|Te|} \tag{2}$$

A third metric was also proposed by the same authors to include the specific position of each document into the evaluation: Expected Normalised Error Reduction (ENER) exploits the probability of a human expert inspecting n documents ($P_s(n)$),

$$\mathrm{ENER}_p = \sum_{n=1}^{|Te|} P_s(n) \cdot \mathrm{NER}_p(n) \tag{3}$$

$P_s(n)$ can follow different probability distributions, Berardi *et al.* [1] suggested the definition shown below, where p models the probability of the next document to be inspected,

$$P_s(n) = \begin{cases} p^{n-1} \cdot (1-p) & \text{if } n \in \{1, ..., |Te| - 1\} \\ p^{n-1} & \text{if } n = |Te| \end{cases} \tag{4}$$

The value of p can be defined as a function of the expected ratio (ξ) of documents being manually classified. $p = \frac{1}{\xi \cdot |Te|}$. Therefore, p is computed for different expected ratios of manually classified documents. Extended information about this evaluation can be found in Berardi *et al.* [1].

Table 1. DDF Levels. c_i represents a class, d a document, and s the classifier's score for each document-class pair. $t(c_i)$ is the threshold for c_i, and $q(c_i)$ is the estimated quality for c_i.

Evidence		Class		Aggregation	
ϵ; given $d, c_i, t(c_i)$		γ; given $d, c_i, t(c_i), \epsilon$		α; given $\gamma(\epsilon, d, \cdot)$	
S	$s(d, c_i)$	**A**	$\epsilon(d, c_i)$	**M**	$\max_{c_i \in C} \gamma(\epsilon, d, c_i)$
A	$\ln(1 + \lvert s(d, c_i) - t(c_i)\rvert)$	**P**	$\begin{cases} \epsilon(d, c_i) & \text{if } s(d, c_i) \geq t(c_i) \\ 0 & \text{otherwise} \end{cases}$	**A**	$\text{avg}_{c_i \in C} \gamma(\epsilon, d, c_i)$
R	$\ln(1 + \frac{\lvert s(d,c_i) - t(c_i)\rvert}{t(c_i)})$			**W**	$\text{avg}_{c_i \in C} q(c_i) \cdot \gamma(\epsilon, d, c_i)$

3 Document Difficulty Framework for SATC

The Document Difficulty Framework (DDF) is a family of document certainty metrics within the context of TC. DDF extends and generalises the principles we explained in previous research [8], exploiting the classification scores and the threshold values within a classifier-independent framework to compute the document certainty. This computation is divided into three different levels, inspired by the comparison of multi-label AL metrics by Esuli *et al.* [4]: evidence, class and aggregation. The **evidence** level computes the confidence value for each document and category, using their classification score and the class threshold. The **class** level specifies which classes are to be considered in the final aggregation. The **aggregation** level combines the filtered confidence levels, producing a document-based certainty.

DDF is based on the composition of the three transformation functions, one for each level, where ϵ represents the evidence, γ the class, and α the aggregation level,

$$\text{certainty}(d) = \alpha\left(\{\gamma(\epsilon(d, \cdot))\}\right) \tag{5}$$

A method within the framework consists in a specific strategy for each level. Table 1 summarises the different candidates analysed herein for each DDF levels. Each method is represented as the concatenation of three letters, representing the strategy followed in each one of the levels. For instance, following Table 1, the difficulty measured by the APA variation is defined as follows,

$$\text{certainty}(d) = \text{avg}_{c_i \in C : s(d, c_i) \geq t(c_i)} \ln(1 + |s(d, c_i) - t(c_i)|) \tag{6}$$

This method considers absolute distance for evidence function (A), only for positive classes (P), and computing the final aggregation as the average topic confidence (A).

3.1 Evidence Level

The evidence level is responsible for computing a confidence value for a document-topic pair. It is modelled as a function $\epsilon \in \mathcal{E} : D \times C \rightarrow [0, \infty]$, where D denotes the set of documents, and C the classes. Three candidates are considered for this level: **score** (S), **absolute** difference (A) and **relative** difference (R).

The first strategy (S) follows the same principle as relevance sampling [7], exploiting the score obtained by the classifier. It assumes that the higher the value the more relevant the score. Therefore, classes with higher scores are the ones with more certainty.

The second method (A) exploits the score and the threshold, assuming that larger distances imply lower uncertainty and higher chance that the document is correctly classified. This assumption is similar to uncertainty sampling [7]. A logarithmic function is applied to limit the effect of very large differences.

The last method (R) applies the same principles as the difference approach. However, it uses a relative difference instead of the absolute value. The rationale is that the absolute distances can be misleading. For instance, a distance of 0.2 would be much more important if the threshold is 0.05 than if it is 0.6.

3.2 Class Level

The class level behaves as a filter, selecting whether to exploit the certainty of a specific label, and hence, if it will be available at the next aggregation step or not. It is defined as a function $\gamma \in \varGamma : \mathcal{E} \times D \times C \rightarrow [0, \infty]$, where a composition with an element $\epsilon \in \mathcal{E}$ would be applied. Two candidates are considered for this level: **all** (A) and **positive** (P).

The first strategy (A) consists on not applying any filtering, hence considering all the confidence scores for a specific document.

The second method (P) selects the classes for which the classification score is higher or equal than the threshold. These are the classes which will be assigned to the document if automatic classification is applied. This strategy aims to focus the difficulty computation on the positive labels. In TC, the positive labels are more representative that the negative ones due to the fact that the number of positive classes for a specific document is usually much smaller than the number of negative ones. For example, the average number of classes per document in `Reuters-21578` is 1.24, while the number of classes is 90. This approach assumes that if all classes are observed, the document certainties are somehow diluted because most of the documents will obtain a high confidence that do not belong to a large subset of the classes.

3.3 Aggregation Level

When multi-label data is used, a certainty value per document has to be provided, since the ranking of the labels can not be used to select nor to rank documents. For example, even if 90% of the most certain labels are selected, it is impossible to decide which documents should be automatically classified. For this reason, the filtered evidence per class should be combined into a single certainty metric for each document. This level is defined as a function $\alpha \in \mathcal{A} : \{\varGamma\} \times D \rightarrow [0, \infty]$. Typically, it will be applied to the set of possible functions $\gamma \in \varGamma$, one for each class $c_i \in C$. This is equivalent, taken a document d and an evidence level function ϵ as inputs, to the set $\varGamma(\gamma, \epsilon, d) = \{\gamma(\epsilon, d, c_i) : c_i \in C\}$. It should be noted that a one-to-one relation exists between the set of classes used in the definition of $\varGamma$ and $\varGamma$ itself, and thus, this notation could be simplified as in Table 1. Three candidates are considered for this level: **maximum** (M), **average** (A) and **weighted** (W).

The first method (M) selects the most certain class for each document. The goal is to rank higher documents with at least one class correctly classified. This is specially important for collections with a low number of classes per document.

The second method (A) averages the confidence values for the filtered classes, providing a general estimation of how certain the class labels are.

The third method (W) uses an averaged weighted linear combination (WLC), based on the quality estimation per class. Classes with low expected quality are weighted less, because even if their assignation seems certain, it is likely to be a misclassification. The estimated quality values are obtained in the cross-validation phase.

4 Results and Discussion

4.1 Experimental Set-Up

The quality of the certainty algorithms for SATC is evaluated using ER and ENER [1]. ER is plotted with different percentages of the collection being manually classified, while the ENER metric provides values to directly compare the quality achieved by DDF methods with other state of the art approaches. Two traditional TC collections (`Reuters-21578` and `20-newsgroups`) are used:

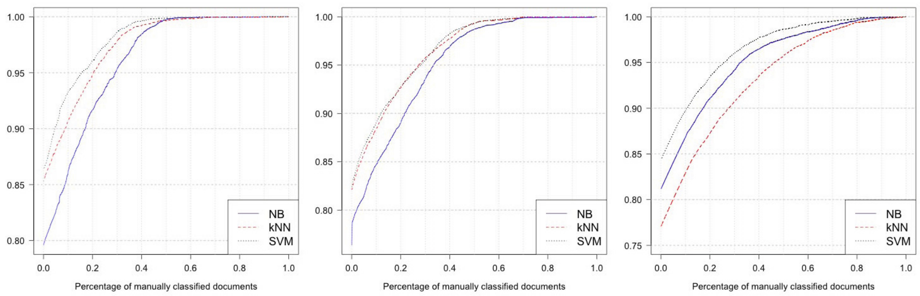
`20-newsgroups` is a collection of approximately $20,000$ newsgroup documents and 20 classes, with almost uniform distribution of documents over classes (Obtained from http://people.csail.mit.edu/jrennie/20Newsgroups/). The split for the collection is based on time as it is suggested. Cross-posting emails have not been considered. This collection has been selected to observe the behaviour of DDF with a single label collection.

`Reuters-21578` contains structured information about newswire articles that can be assigned to several classes. Two variations of the "ModApte" split are used. `Reuters-21578` uses only documents that belong to classes with at least one training and one test document. As a result, there are 7770/3019 documents for training and testing, observing 90 classes with a highly skewed distribution over classes (same as Yang *et al.* [12]). On the other hand, `Reuters-21578-115` uses documents belonging to classes with at least one training or testing document. This configuration has 9603/3299 documents for train and test respectively, and 115 classes. `Reuters-21578-115` allows a direct comparison with the results presented by Berardi *et al.* [1].

Three different families of classifiers have been used, namely Naive Bayes (Weka [5] implementation), SVM using LibSVM [3], and our own implementation of k-NN. Documents are represented using *ltc* [2] and χ^2 is used for feature selection: 5,000 features for NB and k-NN for `20-newsgroups`, and 3,000 for the others. SVM uses 10,000 features (based on full automatic classification experiments). Stemming and stop-words removal have been applied. SCutFBR$_{.1}$ [12] thresholding strategy is applied with an *fbr* value of 0.3 and a 5-fold cross-validation process, optimising micro-average-F$_1$. The SVM scores are obtained by running LibSVM with the option (-b 1).

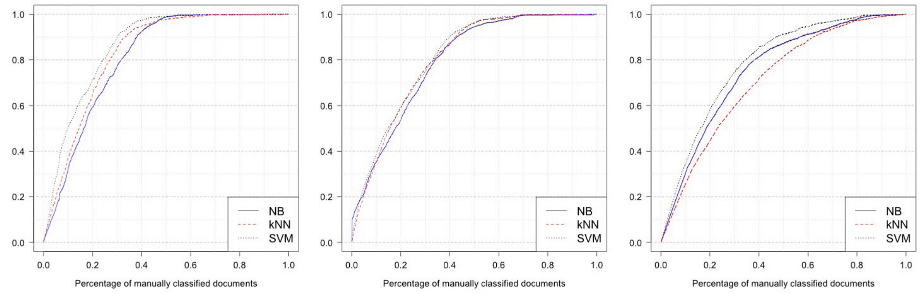
4.2 Results

Figure 1 shows the absolute quality (micro-averaged F$_1$) depending on the percentage of manually classified documents. Due to clarity, only the best DDF metric per collection is shown. It illustrates how the best DDF metrics achieve high quality levels, while manually assigning a small subset of the collections. For instance, for `Reuters-21578`,



(a) `Reuters-21578` SPW (b) `Reuters-21578-115` RAW (c) `20-newsgroups` SPW

Fig. 1. Micro-averaged F_1 evaluation for different ratios of manually classified documents



(a) `Reuters-21578` SPW (b) `Reuters-21578-115` RAW (c) `20-newsgroups` SPW

Fig. 2. Error Reduction (based on micro-averaged F_1) for different ratios of manually classified documents

micro-average F_1 of more than 95% can be achieved with as few as 20% of the documents manually classified. Furthermore, they also show that perfect quality is achieved with approximately 50 and 60% of documents manually classified for `Reuters-21578` and `Reuters-21578-115` respectively. `20-newsgroups` appears to be a more challenging collection for SATC. Perfect quality is only achieved only after 80% of documents are inspected by experts. The main reason for this seems to be the uniformed distribution of documents over classes and the high similarity between some of the classes. The best performing model in all cases is SVM, while k-NN is the second best algorithm, despite the fact that it performs poorly when applied to `20-newsgroups`.

ER evaluation is shown in Figure 2. Although the best model from this perspective is SVM, it illustrates almost overlapped curves for all different classifiers, specially for `Reuters-21578-115`. This result strongly supports the generalisation of DDF metrics. Tables 2-4 allow to directly compare different SATC methods. They show the ENER quality evaluation for all the DDF candidates, with different percentages of the documents expected to be manually classified (ξ). The best performing models presented in the literature are chosen as baselines: The baselines for `Reuters-21578-115` are the results reported by Berardi *et al.* [1] for their Utility-Theoretic method (UT). The method

Table 2. 20newsgroups ENER evaluation wrt. the ratio of manually classified docs. Best results per model in bold, best overall result also underlined. Increment (%) wrt. RPA between brackets.

	NB			kNN			SVM		
	0.05	0.1	0.2	0.05	0.1	0.2	0.05	0.1	0.2
RPA	.097 (0)	.165 (0)	.230 (0)	.076 (0)	.129 (0)	.185 (0)	.121 (0)	.194 (0)	.251 (0)
SAA	-.014 (-114)	-.020 (-112)	-.029 (-113)	.039 (-48)	.059 (-55)	.074 (-60)	.044 (-64)	.046 (-77)	.037 (-85)
SAM	.090 (-7)	.149 (-9)	.207 (-10)	.072 (-5)	.118 (-8)	.161 (-13)	.119 (-1)	.195 (1)	.265 (6)
SAW	.016 (-84)	.032 (-80)	.061 (-74)	.040 (-46)	.062 (-52)	.081 (-56)	.012 (-90)	.033 (-83)	.069 (-72)
SPA	.097 (-0)	.164 (-1)	.227 (-1)	.075 (-1)	.126 (-2)	.176 (-5)	.120 (-0)	.197 (2)	.267 (6)
SPM	.096 (-1)	.157 (-5)	.214 (-7)	.075 (-1)	.123 (-5)	.166 (-10)	.120 (-1)	.196 (1)	.266 (6)
SPW	.097 (0)	.165 (0)	.230 (-0)	.075 (-0)	.127 (-1)	.180 (-3)	**.121** (1)	**.198** (2)	**.268** (7)
AAA	.095 (-2)	.157 (-5)	.214 (-7)	.062 (-18)	.108 (-17)	.159 (-14)	.119 (-2)	.194 (0)	.260 (4)
AAM	.059 (-39)	.095 (-43)	.137 (-41)	.047 (-38)	.076 (-41)	.111 (-40)	.079 (-35)	.136 (-30)	.200 (-20)
AAW	.094 (-3)	.155 (-6)	.213 (-8)	.057 (-24)	.098 (-24)	.145 (-21)	.117 (-3)	.193 (-1)	.259 (3)
APA	.098 (0)	.166 (1)	.233 (1)	.076 (0)	.129 (0)	.185 (0)	.121 (0)	.197 (2)	.262 (4)
APM	.097 (-1)	.161 (-2)	.222 (-4)	.075 (-0)	.126 (-3)	.173 (-6)	.121 (-0)	.196 (1)	.261 (4)
APW	**.098** (1)	**.167** (1)	**.234** (2)	**.076** (0)	**.130** (1)	**.187** (1)	.121 (0)	.198 (2)	.265 (6)
RAA	.089 (-9)	.151 (-9)	.210 (-9)	.061 (-20)	.107 (-17)	.161 (-13)	.119 (-1)	.195 (1)	.260 (4)
RAM	.061 (-37)	.106 (-36)	.162 (-30)	.044 (-41)	.081 (-37)	.126 (-32)	.092 (-24)	.156 (-20)	.216 (-14)
RAW	.089 (-9)	.150 (-9)	.209 (-9)	.056 (-25)	.099 (-24)	.149 (-19)	.118 (-3)	.193 (-0)	.260 (3)
RPM	.096 (-1)	.159 (-4)	.217 (-6)	.075 (-0)	.126 (-2)	.174 (-6)	.120 (-0)	.193 (-0)	.250 (-0)
RPW	.098 (0)	.166 (1)	.233 (1)	.076 (0)	.129 (0)	.186 (1)	.121 (0)	.195 (1)	.256 (2)

we introduced in [8], modelled as RPA within the DDF framework, is used as baseline for the other two collections. UT quality for `20-newsgroups` and `Reuters-21578` was not reported by Berardi *et al.* [1]. In almost all cases, the performance of DDF is higher when SVM is used, instead of NB or k-NN. For `Reuters-21578-115`, several DDF metrics outperform both baselines (UT and RPA), being SAW (with NB), and RAW (with SVM) the best performers. The improvements are as high as 14 and 50% with respect to UT and RPA, respectively. In addition, RAW with NB also outperforms UT in some cases. All collections confirm the quality of DDF, with several candidates outperforming our previously proposed method (RPA) [8]. For `20-newsgroups`, there is almost not difference between the performance of candidates applying average aggregation and those applying weighted aggregation (e.g., APA vs APW). The main reason for this is that the classes quality are very similar. Furthermore, although no one of the best candidates includes the aggregation based on the maximum confidence (surprisingly being a single-label collection), this strategy achieves high quality (i.e. SPM is virtually as good as the best candidate for SVM).

`Reuters-21578` and `Reuters-21578-115` are analysed together as their main difference is the existence of documents without any correct class for the latter. The first observation, in terms of performance, is that models considering positive classes lose their competitiveness against selecting all classes, for `Reuters-21578-115`. The reason is that this strategy was conceived for collections where test documents have at least one correct class, as documents with no classes are assigned a large uncertainty. This also explains the poor performance of our previous method [8], with decreases of more than 50% ENER, both with respect to the best DDF metric and UT. Furthermore, the qualities achieved by the best model in `Reuters-21578` are significantly higher than those for `Reuters-21578-115` (with the exception of those based on NB). This means

Table 3. Reuters21578 ENER evaluation wrt. the ratio of manually classified docs. Best results per model in bold, best overall result also underlined. Increment (%) wrt. RPA between brackets.

	NB			kNN			SVM		
	0.05	0.1	0.2	0.05	0.1	0.2	0.05	0.1	0.2
RPA	.101 (0)	.178 (0)	.245 (0)	.138 (0)	.234 (0)	.320 (0)	.156 (0)	.250 (0)	.321 (0)
SAA	.027 (-73)	.032 (-82)	.026 (-89)	.093 (-33)	.159 (-32)	.232 (-27)	.050 (-68)	.052 (-79)	.042 (-87)
SAM	.089 (-12)	.162 (-9)	.244 (-0)	.103 (-26)	.176 (-25)	.251 (-22)	.171 (10)	.269 (8)	.349 (9)
SAW	**.140** (38)	**.210** (19)	.283 (16)	.099 (-28)	.168 (-28)	.242 (-24)	.157 (0)	.246 (-1)	.321 (0)
SPA	.103 (2)	.184 (4)	.267 (9)	.132 (-5)	.220 (-6)	.304 (-5)	.193 (24)	**.297** (19)	.376 (17)
SPM	.094 (-8)	.166 (-6)	.248 (1)	.117 (-15)	.187 (-20)	.258 (-19)	.171 (9)	.270 (8)	.351 (9)
SPW	.104 (2)	.185 (4)	.269 (10)	.135 (-2)	.226 (-3)	.311 (-3)	.192 (23)	.297 (19)	**.376** (17)
AAA	.105 (4)	.183 (3)	.265 (8)	.159 (15)	.233 (-0)	.302 (-6)	.175 (12)	.262 (5)	.337 (5)
AAM	.055 (-45)	.093 (-48)	.130 (-47)	.072 (-48)	.132 (-44)	.207 (-35)	.041 (-74)	.074 (-71)	.135 (-58)
AAW	.081 (-20)	.157 (-11)	.242 (-1)	**.169** (22)	.244 (4)	.314 (-2)	.157 (0)	.248 (-1)	.326 (2)
APA	.105 (4)	.192 (8)	.277 (13)	.139 (1)	.235 (1)	.323 (1)	.187 (19)	.293 (17)	.372 (16)
APM	.096 (-5)	.174 (-2)	.257 (5)	.122 (-11)	.195 (-16)	.267 (-16)	.164 (5)	.262 (5)	.342 (7)
APW	.105 (4)	.193 (8)	.280 (14)	.141 (2)	.240 (3)	**.329** (3)	.187 (20)	.294 (18)	.374 (16)
RAA	.115 (14)	.189 (6)	.267 (9)	.138 (-0)	.229 (-2)	.312 (-2)	.171 (9)	.260 (4)	.334 (4)
RAM	.082 (-20)	.135 (-24)	.178 (-27)	.082 (-41)	.156 (-33)	.235 (-27)	.119 (-24)	.192 (-23)	.250 (-22)
RAW	.126 (24)	.205 (16)	**.285** (17)	.163 (18)	**.251** (8)	.329 (3)	**.211** (35)	.295 (18)	.361 (13)
RPM	.093 (-8)	.150 (-16)	.193 (-21)	.124 (-10)	.197 (-16)	.265 (-17)	.136 (-13)	.203 (-19)	.254 (-21)
RPW	.106 (5)	.190 (7)	.267 (9)	.142 (3)	.241 (3)	.329 (3)	.170 (9)	.270 (8)	.344 (7)

Table 4. Reuters21578_115 ENER evaluation wrt. the ratio of manually classified docs. Best results per model in bold, best overall result also underlined. Increment (%) wrt. Utility Theoretic (UT) Ranking between brackets (Berardi *et al* [1]).

	NB			kNN			SVM		
	0.05	0.1	0.2	0.05	0.1	0.2	0.05	0.1	0.2
UT	.145	.221	.285	.145	.221	.285	.145	.221	.285
SAA	.014 (-90)	.017 (-92)	.012 (-96)	.064 (-56)	.116 (-48)	.181 (-36)	.049 (-66)	.052 (-77)	.046 (-84)
SAM	.131 (-9)	.181 (-18)	.240 (-16)	.062 (-57)	.118 (-46)	.190 (-33)	.121 (-17)	.201 (-9)	.279 (-2)
SAW	**.184** (27)	**.241** (9)	**.295** (4)	.071 (-51)	.125 (-43)	.191 (-33)	.128 (-12)	.203 (-8)	.272 (-4)
SPA	.055 (-62)	.125 (-43)	.209 (-27)	.065 (-55)	.138 (-37)	.224 (-21)	.096 (-34)	.190 (-14)	.280 (-2)
SPM	.047 (-68)	.107 (-52)	.187 (-34)	.055 (-62)	.113 (-49)	.187 (-34)	.086 (-41)	.173 (-22)	.261 (-9)
SPW	.055 (-62)	.125 (-43)	.209 (-27)	.066 (-55)	.141 (-36)	.228 (-20)	.094 (-35)	.189 (-14)	.279 (-2)
AAA	.119 (-18)	.168 (-24)	.226 (-21)	.135 (-7)	.198 (-10)	.261 (-8)	.101 (-31)	.177 (-20)	.259 (-9)
AAM	.009 (-94)	.019 (-91)	.030 (-90)	.054 (-63)	.105 (-53)	.169 (-41)	.040 (-72)	.082 (-63)	.151 (-47)
AAW	.028 (-81)	.083 (-63)	.160 (-44)	**.144** (-1)	.208 (-6)	.271 (-5)	.074 (-49)	.156 (-30)	.245 (-14)
APA	.055 (-62)	.127 (-43)	.211 (-26)	.067 (-54)	.148 (-33)	.239 (-16)	.094 (-35)	.191 (-14)	.283 (-1)
APM	.047 (-68)	.107 (-52)	.184 (-35)	.058 (-60)	.120 (-46)	.195 (-32)	.082 (-44)	.168 (-24)	.258 (-9)
APW	.055 (-62)	.127 (-42)	.213 (-25)	.068 (-53)	.149 (-32)	.241 (-15)	.093 (-36)	.190 (-14)	.283 (-1)
RAA	.160 (10)	.209 (-5)	.262 (-8)	.110 (-24)	.188 (-15)	.266 (-7)	.123 (-15)	.197 (-11)	.267 (-6)
RAM	.086 (-41)	.125 (-44)	.158 (-45)	.058 (-60)	.120 (-46)	.193 (-32)	.092 (-37)	.152 (-31)	.211 (-26)
RAW	.171 (18)	.225 (2)	.281 (-1)	.138 (-5)	**.214** (-3)	**.285** (-0)	**.165** (14)	**.238** (8)	**.302** (6)
RPA	.045 (-69)	.101 (-54)	.169 (-41)	.064 (-56)	.142 (-36)	.233 (-18)	.083 (-43)	.170 (-23)	.257 (-10)
RPM	.037 (-75)	.066 (-70)	.097 (-66)	.057 (-61)	.117 (-47)	.191 (-33)	.072 (-51)	.139 (-37)	.205 (-28)
RPW	.054 (-63)	.121 (-45)	.198 (-30)	.066 (-54)	.147 (-33)	.240 (-16)	.085 (-41)	.178 (-19)	.269 (-6)

Table 5. Average ENER evaluation for DDF patterns and xi=0.1

Collection	Model	S**	A**	R**	*A*	*P*	**A	**M	**W
20newsgroups	NB	.108	**.150**	.149	.108	**.163**	.130	.138	**.139**
	kNN	.102	.111	**.112**	.090	**.127**	**.110**	.108	.108
	SVM	.144	.186	**.188**	.149	**.196**	.170	**.179**	.168
Reuters21578_115	NB	.133	.105	**.141**	**.141**	.112	.125	.101	**.154**
	kNN	.125	.155	**.155**	**.155**	.135	.155	.116	**.164**
	SVM	.168	.161	**.179**	.162	**.176**	.163	.153	**.192**
Reuters21578	NB	.157	.165	**.174**	.152	**.179**	.160	.147	**.190**
	kNN	.189	.213	**.218**	.194	**.219**	.218	.174	**.228**
	SVM	.238	.239	**.245**	.211	**.271**	.236	.211	**.275**

that the addition of documents without correct classes makes the SATC problem more complex to solve, or at least that DDF metrics are less suited for this type of datasets.

Results also show that SAA (and SAW for 20-newsgroups because of the similar qualities per class) is only suited for classifiers that do not normalise the scores per document. SAA performs as a random ranker for this type of classifiers which include the versions of NB and SVM presented on this paper. If the classification scores are normalised per document ($\sum_{c_i \in C} s(d, c_i) = 1$), SAA produces the same difficulty, independently of the document. Other very poor metric is AAM, because the highest confidence based on difference is usually based on a very low (or even zero) score. Therefore, the certainty computation will be uniquely based on this information.

Table 5 summarises the average quality for candidates sharing two strategies, with $\xi = 0.1$ (arbitrarily chosen). For instance, S** averages the error reduction for every variation using positive labels (this is, it encapsulates information about SAA, SAM, SAW, SPA, SPM, and SPW). This analysis provides information about which strategies are better for each level in different conditions, and it helps to understand some of the previously reported results from a general perspective. For the evidence level, the best strategy is the relative difference, independently of classifier and collection. This result confirms our previous assumptions made for the RPA method [8]. The class level illustrates that the selection of positive classes achieves good quality, as long as the assumption that all the documents have at least one correct class is correct. Otherwise, all classes should be considered. The aggregation level shows that the exploitation of quality estimation outperforms the other strategies for Reuters-21578 and Reuters-21578-115. All strategies perform similarly for 20-newsgroups.

5 Conclusions and Future Work

SATC represents a largely unexplored task within TC which is critical in environments where high quality classification is needed, but resources are limited. Its main goal is to achieve high quality with minimum human effort, minimising the potential cost. This research introduces DDF, a document certainty framework based on classification scores and class thresholds, and its application to SATC.

DDF generalises several methods by abstracting three different levels, specifying how to manipulate the scores and thresholds to obtain a document certainty measure.

Results show that DDF metrics achieve virtually perfect classification with as low as 50% of documents being classified. SVM is the best classifier for DDF and RAW is its best overall variation, with the exception of Reuters-21578-115, where NB with the SAW strategy is the best alternative. DDF outperforms all the previously proposed methods in the literature for SATC. The strategy analysis shows that the best models should include a relative difference of scores, and the exploitation of estimated class quality. In addition, observing only the positive classes for a document achieves better quality, but only if all documents belong to at least one class.

Future work will provide deeper analysis of the combination between difference strategies, as well as their behaviour for different ratios of the collection. The combination of DDF and the Utility-Theoretic Ranking method, and the combination of DDF values as features for a meta-ranker are also interesting lines of research.

References

1. Berardi, G., Esuli, A., Sebastiani, F.: A utility-theoretic ranking method for semi-automated text classification. In: SIGIR (2012)
2. Buckley, C., Salton, G., Allan, J.: The effect of adding relevance information in a relevance feedback environment. In: SIGIR (1999)
3. Chang, C.-C., Lin, C.-J.: LIBSVM: A library for support vector machines. ACM Transactions on Intelligent Systems and Technology (2011)
4. Esuli, A., Sebastiani, F.: Active learning strategies for multi-label text classification. In: Boughanem, M., Berrut, C., Mothe, J., Soule-Dupuy, C. (eds.) ECIR 2009. LNCS, vol. 5478, pp. 102–113. Springer, Heidelberg (2009)
5. Hall, M., Frank, E., Holmes, G., Pfahringer, B., Reutemann, P., Witten, I.H.: The weka data mining software: an update. SIGKDD Explor. Newsl (2009)
6. Larkey, L.S., Croft, W.B.: Combining classifiers in text categorization. In: SIGIR (1996)
7. Lewis, D.D., Gale, W.A.: A sequential algorithm for training text classifiers. In: SIGIR (1994)
8. Martinez-Alvarez, M., Yahyaei, S., Roelleke, T.: Semi-automatic document classification: Exploiting document difficulty. In: Baeza-Yates, R., de Vries, A.P., Zaragoza, H., Cambazoglu, B.B., Murdock, V., Lempel, R., Silvestri, F. (eds.) ECIR 2012. LNCS, vol. 7224, pp. 468–471. Springer, Heidelberg (2012)
9. Sebastiani, F.: Machine learning in automated text categorization. ACM Comput. Surv. (2002)
10. Tong, S., Koller, D.: Support vector machine active learning with applications to text classification. J. Mach. Learn. Res. (2002)
11. Yang, B., Sun, J.-T., Wang, T., Chen, Z.: Effective multi-label active learning for text classification. In: SIGKDD (2009)
12. Yang, Y.: A study on thresholding strategies for text categorization. In: SIGIR (2001)
13. Yang, Y., Liu, X.: A re-examination of text categorization methods. In: SIGIR (1999)