An Empirical Study of Active Learning for Text Classification

Stamatis Karlos[a,c], Nikos Fazakis[b], Sotiris Kotsiantis[c], Kyriakos Sgarbas[b]

[a]Technical Educational Institute of Western Greece, Department of Computer Engineering Informatics, Antirrio 30200, Greece
[b]University of Patras, Department of Electrical Engineering and Computer Science, Rion 26504, Greece
[c]Educational Software Development Laboratory (ESDLab), Department of Mathematics, University of Patras, Greece

Abstract

Text categorization or better text classification has recently attracted the interest of several researchers, since the amount of generated documents on daily basis is vast and on many situations their manipulation is infeasible without using any appropriate Machine Learning tools. Several variants of real-life applications belong to this field and much research has been made the last two decades over them. However, default learning methods do not exploit uncategorized files which are in abundance on several fields. Thus, new learning schemes are exploited for boosting learning performance of supervised algorithms. Active Learning is such a representative example, incorporating both labeled and unlabeled data and integrating human's expertise knowledge with the obtained predictions by supervised learners. In this work, four learners are compared under two different Active Learning approaches against random sampling, examining the efficacy of annotating unlabeled documents that verify specific queries. Classification error has been recorded for two different public provided datasets highlighting the improved learning behavior of using specific queries instead of random sampling approach, under the existence of a really small portion of the initial data.

1. Introduction and related works

Generating textual data or related documents for various reasons, such as recording personal opinions, publishing articles that either describe situations or express agreement/disagreement about a topic of interest, communicating either with analog means or through social media, has been a standard mechanism directly connected with human's nature over several aspects of daily life. Despite the fact that the volume of used images and videos has dramatically increased the last years – being favored by faster and more reliable communication networks and the chance of handling large amount of multimedia data even on mobile devices – the importance of the simple text format has not gotten subdued or it has still been maintained as the exclusive way of serving many applications.

Since Machine Learning (ML) field and its products have been employed by numerous applications for exploiting the assets of several scientific domains like Computer Science, Statistical Learning and Artificial Intelligence, trying to predict qualitative or quantitative variables through mining hidden patterns or unwrapping complex relationships between the provided features, its integration with Text Mining (TM) [1] was inevitable. Some of the most representative examples that have been recently raised and are related with the field of TM are: i) Sentimental analysis, ii) topic based categorization, iii) spam filtering and iv) authorship detection.

The most usual term for this kind of tasks is Text Classification/Categorization (TC) [2] as it has been established in the literature. Meanwhile, some modifications have to be made for matching the well-structured theory and algorithms of ML with the inherent nature of textual data. Having collected *n* documents $\{d_1, d_2, \dots d_n\}$, symbolized as *D*, which are described through a set of *k* predefined classes $\{c_1, c_2, \dots c_k\}$, referring to it with term *Class*, each object of *D* has to belong to exactly one class. If more than one class correspond to any instance then Multilabel [3] TC theory is applied. Thus, the ambition of TC task is to approximate a function $\Phi: D\ x\ C\ \rightarrow \{True, False\}$, as it is defined in [4], which would provide the appropriate matching between documents and classes. To be more specific, when $\Phi\left(d_i, c_j\right) = True$, then the *i-th* document is considered as a positive instance for *j-th* class. The corresponding feature set of a collection of documents *D*, or of a corpus as it is also mentioned in the literature, is formatted by extracting each met word and assigning to it a weight that stems from the measurement of its frequency, according to the default scenario. However, other well-known strategies could be followed, leading to further modifications (e.g. “bag of words” assumption [5]). Hence, dimensions of matrix *D* are symbolized as $t\ x\ f$, where:

1. *t* depicts the cardinality of contained documents, or better to refer as instances hereafter
2. *f* counts the different features, as these were just described.

Although term-frequency-inverse document frequency (tf-idf) is a widely accepted weighing function for TC tasks, its efficacy is poor when only a small number of labeled instances is available. Much research has been made for facing these cases, as in [6] where the most discriminative pair of words are found and an appropriate structure vector is extracted using Latent Dirichlet Allocation (LDA) as probabilistic model. More recently, LDA approach has been also used as a three-level hierarchical Bayesian model for tackling with TC task [7]. The main idea was to be used as a dimensionality reduction technique, under suitable underlying generative probabilistic semantics that are harmonized with the type of data that it models. It actually performed encouraging results reducing at the same time the feature space almost by 99%. Arguably, the representation of feature set plays cardinal role inside a TC task. Besides the more generic techniques that are also applied in other ML datasets, such as feature transformation or reduction and holding the top-rated features depending on relative information metrics, more oriented to the structure of text corpus pre-process methods have been developed. Tokenization, stop-word removal and stemming are probably the most well-known and a recent work that highlights the impact of such editing techniques before the assessment of textual datasets by the basic kernel of ML tools is [8]. There, a large amount of combinations of the most important pre-processing methods was evaluated for examining their influence over two different domains and languages of text files.

Essentially, all the previously referred concepts and editing procedures are designed, or modified appropriately, so as to correspond to the text nature of corresponding corpus, but do not keep pace with the vast amounts of data that current data scientists have to manipulate. Since the annotation of text documents is a really slow process analog to the size and/or the characteristics that the asked human expertise has to detect, the operation of the supervised learning methods with classifiers that operate exclusively under the existence of categorized data seems a not efficient approach [9]. On the contrary, Active Learning (AL) methodology supports the mining of useful information through non-annotated instances which are selected under some specific criteria. Furthermore, in order to increase its confidence and reduce the generalization error, AL integrates inside its learning process the human factor. Thus, the role of ML learners is to search into uncategorized instances for discriminating the most representative and/or informative of them per each iteration, in order to acquire the appropriate class labels through asking the human expertise, boosting in this way the total knowledge over the whole dataset [10]. Such evidence-based frameworks have been proven really beneficial for tackling TC problem [11].

The main contribution of this work is the examination of several learners under some AL schemes for observing their efficacy without using any complex or computationally expensive pre-process stage, while just a small sample of the collected data is assumed to be initially categorized. The remainder of this paper is organized as follows: in Section 2 the basic details of AL methodology are recorded. Section 3 contains the description of the used data, the query strategies that were exploited and some more technical details related with the upcoming experimental procedure, while in Section 4 the conducted comparisons along with illustrative learning curves are placed. Finally, the last section summarizes our conclusions and the proposed future work.

2. Active Learning

Beyond the default categories of supervised and unsupervised learning, new learning schemes related with the domain of ML have come to the foreground the last decades. Semi-Supervised Learning (SSL), Reinforcement Learning, the more recent Deep Learning are the most illustrative, as well as other interactive variants, like Adaptive Learning. The basic orientation that AL follows, contains some of the basic properties or concepts of aforementioned directions, but incorporates two new factors: setting queries and asking human factor for annotating specific instances.

More specifically, AL comprises two different kind of data: labeled (*L*) and unlabeled (*U*). The only characteristic that helps us to discern each other is the absence of the class value from the latter kind. This property is not tackled neither with missing values theory nor with omitting these instances from the learning phase. Therefore, existence of *U* subset constitutes ideally a source of useful information and its exploitation could be declared as the basic mechanism of mining information under AL approach. Although the strategy of exploiting the *U* subset is also closely interwoven with SSL generating automated mechanisms and tools, AL concept is designed towards serving more realistic applications incorporating both human factor and query frameworks inside its training process. In this way, less autonomous tools are produced, since human's supervision is needed, but the benefits from adjusting the manner of obtaining knowledge through such available data by asking appropriate queries leads to production of tools that could be theorized as semi-autonomous with great flexibility over both general and more specified tasks [12].

The number of AL reviews is still small, but the offered works are really instructive [13],[14],[15]. Different aspects of this kind of learning are discussed by their authors, examining either the performance of various query frameworks or the co-operation of more than one classifiers against the default scenario of one classifier. Additionally, matters of how much the learning phase is affected when more than one queries are activated, or how to expand the AL concept over other tasks or procedures, such as implementation of AL in sequences and graphs, are examined in depth. Despite the variety of AL expressions that have demonstrated until now, they all depend on the choice of suitable scenarios for exploiting instances that come from *U* subset, and afterwards, on the implementation of queries that are harmonized with the nature of the data or the given application. According to Settles and its subsequent survey about the concept of AL [16], the following general frameworks could be detected when an AL task is going to take place:

1. *Query Synthesis*, where the instances are scrutinized under the hypothesis that are generated by incorporating attributes of more than one of the original examples. Appropriate information should be provided in this case, describing the ranges of each feature inside the whole dataset, especially for regression tasks or artificial problems.
2. *Stream-based Selective Sampling*, where after having constructed the corresponding learning model based on available dataset *L*, one instance per iteration is extracted - simulating thus the well-known stream data phenomenon – and our model has to decide if this could be proven useful or not. Analog to how strict are the posed conditions that have to be met by the incoming instances, the level of human supervision and its spent effort are adjusted.
3. *Pool-based Sampling*, where the original specification about the existence of two separate subsets *L* and *U* is clearer than the other two cases. Based on the formatted hypothesis through the *L* subset, instances that satisfy better an

objective function, or are placed into decision regions that little information is known about them or even disagreement behaviors from different learners are detected about their labels, are asked to be evaluated and then are inserted into *L* subset for enhancing the learning ability of our model.

All these three approaches have found acceptance in the literature and on real-life situations [17],[18]. Apart from the first case, which is easily discerned because of the mixed nature that its instances are governed, the other two differentiate mainly over the memory limitations on practical level and on the fact that the former is enabled when a new unlabeled instance is found, while the latter demands a more compact structure of the *U* subset for judging the suitability of the examined instances. However, the queries that are asked are similar and the interest of researchers is shifted towards such directions [19].

3. Experimental Methodology

This section is separated into three distinct subparagraphs, following the procedure that was respected before we execute our experiments. A short description of each one's content is provided here: how to find the appropriate text datasets, select the most favoring AL approach based on the properties of the collected datasets and finally select representative learning algorithms so as to examine their efficacy over the field of classifying text data using AL.

3.1. Dataset Description

The used in our work datasets are extracted by public repository [20] and are part of a larger and widely used corpora in the field of TC, which is called 'Reuters-21578 – Distribution 1.0' [21]. The included data was initially collected by Carnegie Group, Inc. and Reuters, Ltd. as a part of their purpose to develop the CONSTRUE text categorization system. This corpus consists of 22 files that each one contains 1000 documents, except for one that contains the remaining 578 files, explaining thus the name of the corpus. The ModApte split has been chosen for our work, which is also reviewed and examined in [22],[4]. This split contains 9603 training and 3299 test documents, respectively. Following the principles of TC, each word plays the role of a feature into the formatted dataset and the classes have been selected to be the different topics that were identified to be discussed into the newswire articles.

Although 135 different topics were totally encountered through all the corpus, only 90 of them were kept, since it was necessary to exist at least one appearance of each topic in both training and test group of documents. A different dataset was then built for each topic, generating 90 binary datasets, where the final class describes the relation of any feature-word with the corresponding topic (Yes/No class values). Moreover, after the standard cleaning phase of stemming and stop-word removal was applied, 9947 distinct terms were detected.

As it concerns the previously referred repository [20], it provides the R10 corpus, which means that only the top ten topics are included, sorted by the cardinality of the documents that were discussed as topics. In order to implement AL experiments and do not review the full size of the R10 collection, we applied a fast 3-cross-validation evaluation method for recognizing the more ambiguous datasets and reached to the point that ACQ and EARN datasets could be proven the most useful for our research. Their properties are reported in the following Table:

Table 1. Quantitative description of examined datasets

Datasets	Train docs	Test docs	Instances (Yes – No)	Features
ACQ	1596	696	12897 (3964 – 8933)	7495
earn	2840	1083	12897 (2369 – 10528)	9500

3.2. Active Learning Queries

Several AL strategies have been recorded in the literature. Although the degrees of freedom that are provided inside them are in abundance – any learner that outputs class probabilities is permitted in the majority of strategies, the number of the included learners into the combining variants is not restricted and any objective function can also be defined for assigning corresponding confidence scores – some general properties are maintained and could guide researchers to wiser options. For example, Uncertainty Sampling (UncS) is characterized by fast enough response but remains too self-confident, without supporting any mechanism that compensates the case of poor provided data. Disagreement methods combine the decisions of any base learners but cannot guarantee reduction of generalization error, while the problem of diversity has also to be tackled. Since the previous strategies belong to Heterogeneity-based models [14], other corresponding groups of learning models could similarly be examined. For example, performance-based models that try to optimize their decisions analog to one or more selected objective functions could facilitate the mining of instances that satisfy more complex criteria. However, their need of much computational resources, renders them us infeasible solutions for applications that time response plays cardinal role. More details could be found in previously referred surveys.

In our case, judging by the large dimensionality of the available datasets, we applied UncS strategy with three different learning options under a Pool-based scenario. Consequently, we search for the instances that belong to the *U* subset and their uncertainty is the largest possible each time. Taking into consideration that our problems are binary, instances that are assigned with probability values close to 0.5 for each class (or to 1/number of classes for multiclass problems) are the most informative for enriching our model's learning ability. Hereafter, we will assume that our problems are all binary and no comments about the multiclass case will be given. Thus, the three most well-known measures that could be used for converting term of uncertainty into an arithmetic form are merged into two distinct in our work, since Small Margin (SM) query is exactly the same with the Least Confident (LC) measure. We will keep the term LC as it is more generic for the rest of our paper. Their corresponding formulas are presented here:

1. *Least Confident* (LC), queries the instances whose posterior probability of satisfying our assumption is nearest to 0.5:

$$x_{LC}^{selected} = \arg\max_{x} 1 - P_{model}(\hat{y}|x) \tag{1}$$

2. *Entropy* (E), queries the instances with low variable information per class:

$$x_{E}^{selected} = \arg\max_{x} H_{model}(Class|x) =$$

$$\arg\max_{x} - \sum_{y \in class} P_{model}(y|x) * \log P_{model}(y|x) \tag{2}$$

Where $\hat{y} = arg \max_{x} P_{model}(y|x)$, *x* is used for describing any example that belongs to the *U* pool, *y* is used as the target class and iterates over all the possible classes, which are encoded to a row-vector with name *Class*. Furthermore, in order to compare their performance with a similar reference method, the default tactic in AL applications is to apply Random Sampling (RS) strategy. Based on this, random indices are produced at each iteration and the instances which match with them are extracted and asked by human expert to be classified. Under this case, a straight decision about the worth of utilizing any metric during the mining of information by the *U* subset can be drawn. It has to be mentioned that it would be unfair to perform comparisons of AL approaches against the behavior of the default supervised model built only on the initial selected population, since AL scheme is an incremental framework that assumes the integration of correctly classified instances per iteration into the *L* subset. Hence, the initial *L* could not provide a better learning view against an augmented dataset which is in fact its superset.

3.3. Experimental setup

Implementation of above referred AL approaches was carried out using the libact [23]: Pool-based Active Learning in Python package. Besides the contained classifiers that are placed to the current version, there is the chance to incorporate any classifier that is supported on Scikit-learn library, one of the greatest collection of ML tools. The only restriction that is posed by UncS is the demand of Probabilistic classifiers, a property that may not be satisfied by the majority of the contained classification algorithms. Nevertheless, it is relative easy to overcome this barrier by inheriting specific methods that facilitate the export of algorithms' decisions under the appropriate Probabilistic requirement.

For fulfilling our experiments, we selected 4 classification algorithms that come from different learning families: Statistical Learning, Decision Trees (DT), Bayesian Learning and Support Vector Machine (SVM) Learners. We did not apply many of them from each category but just one, due to lack of space for representing large volume of comparisons and since our research is in a primary phase over the field of TC under AL concept. Short description of the chosen classifiers per category is placed here:

1. Logistic Regression (LogReg) [24], provided a given L subset, conditional distribution is approximated by optimizing a fit parameters problem. Inside Scikit-learn library, this algorithm performs regularized logistic regression using the 'liblinear' library.
2. ExtraTreesClassifier (ExtraTr) [25], a meta–estimator that fits a number of randomized DTs on various subsets of the original L subset. It also uses averaging technique for improving its accuracy and controlling any over–fitting phenomena.
3. MultiomialNB (MNB) [26], it supports the classical Naive Bayes behavior favoring classification with discrete features and operating both with integer and fractional feature counts.
4. Stochastic Gradient Descent (SGD) [27], a simple yet very efficient approach to discriminative learning of linear classifiers under convex loss functions such as SVMs and LogReg with large acceptance over the context of large-scale learning.

No parameters tuning of the tested classifiers was made, since our ambition still remains to examine the general behavior of combining these algorithms with AL scheme and not to maximize any observed metric. The only comment that has to be reported is the use of 'log' loss function in case of SGD classifier, which returns the probabilistic classifier of the logistic regression as the linear classifiers.

4. Results

Before the presentation of the produced results, some technical details will be provided here. In order to complete our experiments and evaluate our AL methods, we performed the following evaluation process: First of all we split the initial available data to train and test subsets with a split ratio equal to 0.25 for the former. Then, we choose 130 randomly selected instances for formatting the initial labeled population (*L*). Then, 10 iterations were set to take place selecting the 13 highest ranked instances, according to the examined AL approach. Subsequently, at the end of the 10th iteration, a new set *L'* will have been formatted with the double cardinality of the initial *L* subset. Next, the corresponding classification method is built based on *L'* and we apply it on the test set. We repeat this process 5 times and average the classification accuracies.

During our experiments, the human expert has been replaced by a computer 'oracle' that makes no mistakes and reveals the real class label of any asked instance. This means that we assume the availability of correct labels. For the opposite scenario, noisy instances included, alternative solutions could be found in the literature [13]. Next, Table 2 depicts the relative reduction of classification error rate between the initial *L* subset and the final *L'* subset, after having completed all the 10 scheduled iterations for both examined datasets.

Table 2. Achieved relative reduction of classification error rate (%) for tested AL strategies

Dataset	UncS (LC)				UncS (E)			
	LogReg	ExtraTr	MNB	SGD	LogReg	ExtraTr	MNB	SGD
ACQ	68.1	28.3	72.1	54.1	68.1	31.7	72.1	61.1
EARN	73.2	37.4	68.6	58.4	73.2	29.5	68.6	57.2

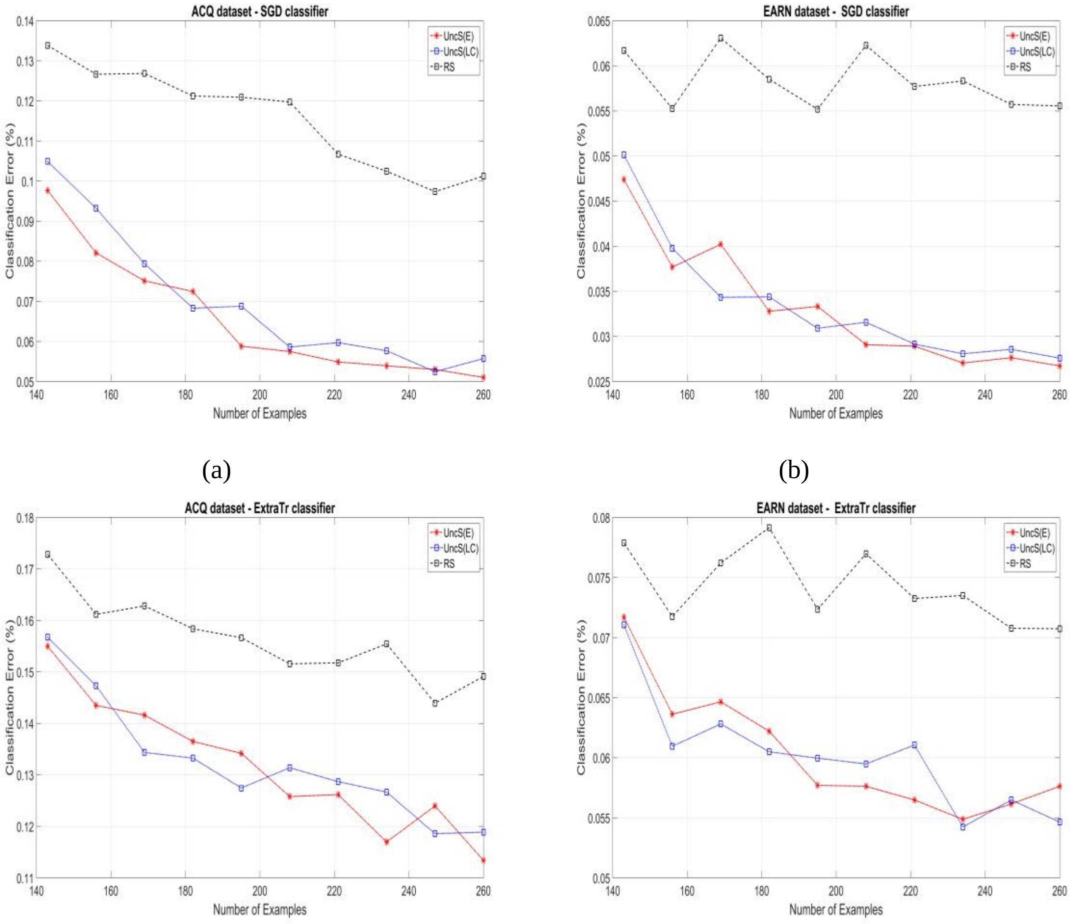
We notice that the best relative reduction was achieved exploiting MNB classifier over both UncS approaches for ACQ dataset (from about 18.3% to 5.1%, which is translated into 72.1% relative reduction), while LogReg scored the best corresponding behavior in case of EARN dataset (from about 9.7% to 2.6%, which is translated into 73.2% relative reduction). Since AL concept is an interactive method, it is also important to illustrate the behavior of the various tested approaches over the executed iterations. Some of the reasons that such a need appears are the comparison of the candidate proposed approaches with the RS strategy along all the learning procedure, the detection of any quality characteristics related with the learning curve (slow or fast converge, possible fluctuations) and observation of local/global minimum or maximum points. In Table 3, the relative error rate reduction for all 4 learners against their corresponding RS strategy is presented. These values have been computed abstracting the error rate during the final iteration (10th) of each base learner from the corresponding achieved error rate using RS strategy, and this result is divided with the error rate of the base learner at the initial iteration (0th). Due to the observed unstable curves that were recorded during the learning process in cases of MNB and LogReg as base learners, and despite the fact that the

Table 3. Achieved relative reduction of classification error rate (%) for tested comparisons

Dataset	UncS (LC) vs RS				UncS (E) vs RS			
	LogReg	ExtraTr	MNB	SGD	LogReg	ExtraTr	MNB	SGD
ACQ	61.5	18.2	72	37.5	61.5	21.5	72	38.2
EARN	53.5	18.4	65.8	42.1	53.5	16	65.8	46.1

improvement of their error rates against their corresponding RS strategy were adequate enough, the most robust learning behaviors were achieved by using SGD and ExtraTr algorithms. This is explained by the poor recorded behavior of the RS strategies of these learners, which in some cases remained unchanged for all the performed iterations. Moreover, the differences using either the LC or the E query were minimal. Thus, the learning curves of the rest two algorithms that happens to be the most representative active learners are illustrated below:

Fig. 1. Learning curves of error rate for SGD classifier (plots a & b) and ExtraTr classifier (plots c & d)



(a)

(b)

(c)

(d)

As it is depicted above, the y-axis contains measurements of classification error (%). In all these four cases, both UncS(E) and UncS(LC) achieved clearly better learning behaviors, since from the beginning of the learning procedure till its end no point was observed to be placed above the corresponding curve of RS approach. Moreover, it is really encouraging the fact that the performance of the two former approaches manage to outreach RS's performance from the initial iteration, although no safe decision could be extracted about which between these two is better. Experiments with several choices of the number of selected instances per iteration, as well as different amounts of initial population should be executed for obtaining a more clear view about the efficacy of queries under AL strategies. However, the produced results using *L* subset whose cardinality equals to about only 1% of the original size of each examined dataset seem to be satisfying enough.

It is also mentioned again that the number of iterations inside AL approaches is just 10, requesting a few instances to be annotated by our human oracle per iteration, preserving the human effort to relatively low levels, compared with other works that exploit less instances over larger number of iterations for achieving more robust behaviors.

5. Conclusions

Summarizing, in this work a brief review about the field of TC and the reasons why schemes that exploit unlabeled instances are really attractive for this field are discussed. The basic properties and assets of AL strategies are also recorded, bridging with a smooth way the gap between default ML methods with this more recent kind of semi-automated tools. Detailed descriptions about our applied experimental procedure were given and comparison of two distinct AL schemes (Uncertainty sampling with Least Confident metric and Entropy measure) with RS approach, that randomly selects instances for augmenting the initial labeled set, using four different classification algorithms were executed. Although only two datasets were assessed, the generated results were encouraging revealing the benefits of incorporating AL strategies during building more robust and accurate classification models. Relative improved error rate values along with learning curves that visualize classification error over each iteration justify our positive attitude towards AL theory and its practical worth over real-life applications.

As it concerns possible enhancements of our work, larger amount of experiments should take place. The parameters that could be tapped are: number of inserted instances per iteration, cardinality of the *L* subset, general tuning of any used classifier either enabling the construction of a validation set or by a typical cross-validation pre-process stage. Moreover, since human effort is demanded during AL concept, reduction of spent effort has to be kept in a high priority during the development of such data mining tools. One direction towards which we should move could be the construction of ensemble learners so as to enhance the decision quality of both the learner that queries instances from the *U* subset and of the evaluator, since more informative mining and more robust behavior could be reassured by such combinations [28]. Another tactic could be the combination of AL and SSL strategies, controlling the trade-off between the participation of human factor and the achieved accuracy, as it has happened with great success to other scientific fields [29],[30].

Our results also enforce the ability of AL scheme to be adopted by commercial applications. The fact that even when the classification accuracy of the supervised model was high enough (90% in case of EARN dataset), after just 10 iterations the achieved accuracy was clearly better, letting us to expect the satisfaction of even stricter specifications that the needs of top rated tools may set. Finally, conduction of experiments with AL strategies that are quite computationally expensive, and were excluded from this work, could provide safer views and more generic conclusions about the most suitable AL strategy related with the TC problem. Insertion of pre-process stages that implement dimensionality reduction could be proven helpful in these cases, especially if numerous learners have to be assessed either individually or under committees.

Lastly, some additional proposals for future work, we would like also to examine the Multi-Label cases exploiting the corresponding AL strategies or employ new proposed methods that also take into consideration the human effort and demand less expertise ability, reducing thus both time and expenses [31]. Meanwhile, the effect of reusability should also tested. Based on this phenomenon, weak learners or learners with better time response are used as annotators during the mining of the most informative instances and different algorithms are used for evaluation [32]. It is evident that the number of possible combinations increases dramatically. Besides the examination of the generated learning curves, new measures for specifying the most suitable combination may be introduced [33].

Acknowledgements

This research has received funding from the European Union's Horizon 2020 research and innovation programme "FrailSafe" under grant agreement No 690140. Moreover, Stamatis Karlos gratefully acknowledges the support of his work by both the General Secretariat for Research and Technology (GSRT) and the Hellenic Foundation for Research and Innovation (HFRI).

References

[1] B. Baharudin, L. H. Lee, and K. Khan, "A Review of Machine Learning Algorithms for Text-Documents Classification," *J. Adv. Inf. Technol.*, vol. 1, no. 1, Feb. 2010.

[2] S. Tong and D. Koller, "Support vector machine active learning with applications to text classification," *J. Mach. Learn. Res.*, vol. 2, pp. 45–66, 2002.

[3] G. Tsoumakas and I. Katakis, *Multi-Label Classification*, vol. 3, no. 3. 2007.

[4] F. Sebastiani, “Machine Learning in Automated Text Categorization,” *ACM Comput. Surv.*, vol. 34, no. 1, pp. 1–47, 2001.

[5] C. D. Manning, P. Ragahvan, and H. Schutze, “An Introduction to Information Retrieval,” *Inf. Retr. Boston.*, no. c, pp. 1–18, 2009.

[6] F. Colace, M. De Santo, L. Greco, and P. Napoletano, “Text classification using a few labeled examples,” *Comput. Human Behav.*, vol. 30, pp. 689–697, 2014.

[7] J. C. Campbell, A. Hindle, and E. Stroulia, “Latent Dirichlet Allocation,” *Art Sci. Anal. Softw. Data*, vol. 3, pp. 139–159, 2015.

[8] A. K. Uysal and S. Gunal, “The impact of preprocessing on text classification,” *Inf. Process. Manag.*, vol. 50, no. 1, pp. 104–112, 2014.

[9] K. Nigam, A. K. Mccallum, S. Thrun, and T. Mitchell, “Text Classification from Labeled and Unlabeled Documents using EM,” *Mach. Learn.*, vol. 39, no. 2/3, pp. 103–134, 2000.

[10] B. Du, Z. Wang, L. Zhang, L. Zhang, W. Liu, J. Shen, and D. Tao, “Exploring representativeness and informativeness for active learning,” *IEEE Trans. Cybern.*, vol. PP, no. 99, pp. 1–13, 2015.

[11] M. Sharma and M. Bilgic, “Evidence-based uncertainty sampling for active learning,” *Data Min. Knowl. Discov.*, vol. 31, no. 1, pp. 164–202, 2017.

[12] G. V. Cormack and M. R. Grossman, “Scalability of Continuous Active Learning for Reliable High-Recall Text Classification,” in *Proceedings of the 25th ACM International on Conference on Information and Knowledge Management - CIKM ’16*, 2016, pp. 1039–1048.

[13] B. Settles, “Active learning literature survey,” *Univ. Wisconsin, Madison*, vol. 52, pp. 55–66, 2010.

[14] C. C. Aggarwal, X. Kong, Q. Gu, J. Han, and P. S. Yu, *Active Learning: A Survey*. 2014.

[15] M. E. Ramirez-Loaiza, M. Sharma, G. Kumar, and M. Bilgic, “Active learning: an empirical study of common baselines,” *Data Min. Knowl. Discov.*, vol. 31, no. 2, pp. 287–313, Mar. 2017.

[16] B. Settles, “Active Learning,” *Synth. Lect. Artif. Intell. Mach. Learn.*, vol. 6, no. 1, pp. 1–114, Jun. 2012.

[17] J. Kranjc, J. Smailović, V. Podpečan, M. Grčar, M. Žnidaršič, and N. Lavrač, “Active learning for sentiment analysis on data streams: Methodology and workflow implementation in the ClowdFlows platform,” *Inf. Process. Manag.*, vol. 51, no. 2, pp. 187–203, Mar. 2015.

[18] L. Feng, Y. Wang, and W. Zuo, “Quick online spam classification method based on active and incremental learning,” *J. Intell. Fuzzy Syst.*, vol. 30, no. 1, pp. 17–27, Aug. 2015.

[19] S.-J. Huang, R. Jin, and Z.-H. Zhou, “Active Learning by Querying Informative and Representative Examples,” *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 36, no. 10, pp. 1936–1949, 2014.

[20] “Download Ohsumed and Reuters, two standard corpora for text classification in ARFF format.” [Online]. Available: https://www.mat.unical.it/OlexSuite/Datasets/SampleDataSets-download.htm. [Accessed: 08-Jul-2017].

[21] “Reuters-21578 Text Categorization Collection.” [Online]. Available: http://kdd.ics.uci.edu/databases/reuters21578/reuters21578.html. [Accessed: 08-Jul-2017].

[22] T. Joachims, “Text Categorization with Suport Vector Machines: Learning with Many Relevant Features,” *Proc. 10th Eur. Conf. Mach. Learn. ECML ’98*, pp. 137–142, 1998.

[23] Y.-Y. Yang, S.-C. Lee, Y.-A. Chung, T.-E. Wu, S.-A. Chen, and H.-T. Lin, “libact: Pool-based Active Learning in Python,” 2015. [Online]. Available: https://github.com/ntucllab/libact.

[24] A. Ng and M. I. Jordan, “On generative vs. discriminative classifiers: A comparison of logistic regression and naive bayes,” *Proc. Adv. Neural Inf. Process.*, vol. 28, no. 3, pp. 169–187, 2002.

[25] P. Geurts, D. Ernst, and L. Wehenkel, “Extremely randomized trees,” *Mach. Learn.*, vol. 63, no. 1, pp. 3–42, 2006.

[26] A. M. Kibriya, E. Frank, B. Pfahringer, and G. Holmes, “Multinomial Naive Bayes for Text Categorization Revisited,” in *AI 2004: Advances in Artificial Intelligence*, 2004, pp. 488–499.

[27] L. Bottou, “Large-Scale Machine Learning with Stochastic Gradient Descent,” *Proc. COMPSTAT’2010*, pp. 177–186, 2010.

[28] G. Wang, J. Sun, J. Ma, K. Xu, and J. Gu, “Sentiment classification: The contribution of ensemble learning,” *Decis. Support Syst.*, vol. 57, no. 1, pp. 77–93, 2014.

[29] W. Han, E. Coutinho, H. Ruan, H. Li, B. Schuller, X. Yu, and X. Zhu, “Semi-supervised active learning for sound classification in hybrid learning environments,” *PLoS One*, vol. 11, no. 9, pp. 1–23, 2016.

[30] M. S. Hajmohammadi, R. Ibrahim, A. Selamat, and H. Fujita, “Combination of active learning and self-training for cross-lingual sentiment classification with density analysis of unlabelled samples,” *Inf. Sci. (Ny).*, vol. 317, no. C, pp. 67–77, Oct. 2015.

[31] S. J. Huang, S. Chen, and Z. H. Zhou, “Multi-label active learning: Query type matters,” *IJCAI Int. Jt. Conf. Artif. Intell.*, vol. 2015–Janua, no. Ijcai, pp. 946–952, 2015.

[32] R. Hu, B. Mac Namee, and S. J. Delany, “Active learning for text classification with reusability,” *Expert Syst. Appl.*, vol. 45, pp. 438–449, Mar. 2016.

[33] K. Tomanek and K. Morik, “Inspecting Sample Reusability for Active Learning.,” *Act. Learn. Exp. Des. AISTATS*, vol. 2011, pp. 169–181, 2011.