On Using Crowdsourcing and Active Learning to Improve Classification Performance

Joana Costa[1], Catarina Silva[1,2], Mário Antunes[1,3], Bernardete Ribeiro[2]

[1] *Computer Science Communication and Research Centre, Polytechnic Institute of Leiria, Portugal*
{joana.costa,catarina,mario.antunes}@ipleiria.pt
[2] *Center for Informatics and Systems, University of Coimbra, Portugal*
{catarina,bribeiro}@dei.uc.pt
[3] *Center for Research in Advanced Computing Systems, University of Porto, Portugal*

***Abstract*—Crowdsourcing is an emergent trend for general-purpose classification problem solving. Over the past decade, this notion has been embodied by enlisting a crowd of humans to help solve problems. There are a growing number of real-world problems that take advantage of this technique, such as Wikipedia, Linux or Amazon Mechanical Turk.**

In this paper, we evaluate its suitability for classification, namely if it can outperform state-of-the-art models by combining it with active learning techniques. We propose two approaches based on crowdsourcing and active learning and empirically evaluate the performance of a baseline Support Vector Machine when active learning examples are chosen and made available for classification to a crowd in a web-based scenario.

The proposed crowdsourcing active learning approach was tested with Jester data set, a text humour classification benchmark, resulting in promising improvements over baseline results.

***Keywords*-Crowdsourcing, Active Learning, Support Vector Machines, Text Classification**

I. Introduction

Crowdsourcing systems enlist a multitude of humans to help solve a wide variety of problems. Over the past decade, numerous systems have appeared on the World-Wide Web, such as `Wikipedia`, `Linux`, `Yahoo! Answers` and `Mechanical Turk` based systems [1].

Crowdsourcing emerged as a new paradigm for using all the emerging available information and opinion shared among users. Hence, this model is capable of aggregating talent, leveraging ingenuity while reducing the costs and time formerly needed to solve problems [2]. Moreover, crowdsourcing is enabled only through the technology of the web, which is a creative mode of user interactivity, not merely a medium between messages and people [2].

In classification scenarios, a large number of tasks must deal with inherently subjective labels and there is substantial variation among different annotators. However, in spite of the attention crowdsourcing has received in fields such as philosophy, linguistics, and psychology, there have been few attempts to create computational models for automatic classification.

On another perspective, active learning designs and analyses learning algorithms that can effectively filter or choose the samples to be labeled by a supervisor (a.k.a. oracle or teacher). The main reason for using active learning is mainly to expedite the learning process and reduce the labeling efforts required by the supervisor [3]. Another strong reason is related to the possibility that each user has to define personal labels and then to build upon a customised learning model that better fits his preferences.

The crowdsourcing active learning framework we propose is a certainty-based method that uses the definition of the most specific informative examples to improve baseline performance, always considering that the number of active examples has to be necessarily small to be classified by the crowd.

The rest of the paper is organized as follows. We start in Section II by describing the background on the learning approaches, namely Support Vector Machines, active learning and crowdsourcing. We then proceed into Section III by presenting the proposed framework for active learning crowdsourcing. Then, in Section IV we introduce the case study of humour classification and the Jester benchmark. In Section V we present and analyze the results obtained. Finally, in Section VI we present the most relevant conclusions and delineate some directions for future work.

II. Learning Approach

In this section we present the background on Support Vector Machines (SVM), active learning and crowdsourcing, which constitute the generic knowledge for understanding the approach proposed in this paper.

A. Support Vector Machines

SVM is a machine learning method introduced by Vapnik [4], based on his Statistical learning Theory and Structural Risk Minimization Principle. The underlying idea behind the use of SVM for classification, consists on finding the optimal separating hyperplane between the positive and negative examples. The optimal hyperplane is defined as

978-1-4577-1676-8/11/$26.00

the one giving the maximum margin between the training examples that are closest to it. Support vectors are the examples that lie closest to the separating hyperplane. Once this hyperplane is found, new examples can be classified simply by determining on which side of the hyperplane they are.

The output of a linear SVM is $u = \mathbf{w} \times \mathbf{x} - b$, where $\mathbf{w}$ is the normal weight vector to the hyperplane and $\mathbf{x}$ is the input vector. Maximizing the margin can be seen as an optimization problem:

$$\begin{aligned} &minimize \quad \frac{1}{2}||\mathbf{w}||^2, \\ &subjected\ to\ \ y_i(\mathbf{w}.\mathbf{x}+b) \geq 1, \forall i, \end{aligned} \tag{1}$$

where $\mathbf{x}$ is the training example and y_i is the correct output for the ith training example. Intuitively the classifier with the largest margin will give low expected risk, and hence better generalization.

To deal with the constrained optimization problem in (1) Lagrange multipliers $\alpha_i \geq 0$ and the Lagrangian (2) can be introduced:

$$L_p \equiv \frac{1}{2}||\mathbf{w}||^2 - \sum_{i=1}^{l} \alpha_i(y_i(\mathbf{w}.\mathbf{x}+b) - 1). \tag{2}$$

In fact, Support Vector Machine (SVM) constitute currently the best of breed kernel-based technique, exhibiting state-of-the-art performance in diverse application areas, such as text classification [5]–[7]. In humour classification we can also find the use of SVM to classify data sets [8], [9].

B. Active Learning

The key idea behind active learning is that a machine learning algorithm can achieve greater accuracy with fewer training labels if it is allowed to choose the data from which it learns. An active learner may pose queries, usually in the form of unlabeled data instances to be labeled by a supervisor [10].

The reason for using active learning is mainly to expedite the learning process and reduce the labelling efforts required by the supervisor, therefore active learning is well-motivated in many modern machine learning problems where data may be abundant but labels are scarce or expensive to obtain [3], [10]. Another strong reason is the possibility of each user define personal labels, thus constructing a customized learning model that better fits his preferences. The customization of a learning model is particularly important in recommendation applications, like movie or book recommendation systems.

Active learning methods can be grouped according to the selection strategy, as being committee-based and certainty-based [11]. In the first group the active examples combine the outputs of a set of committee members, by determining those in which the members disagree the most as the candidates to be labeled [12]. The certainty-based methods try to determine the most uncertain examples and point them as active examples to be labelled. The certainty measure depends on the learning method used.

Active learning has been successfully applied to a large number of computational tasks, such as, text classification, image recognition or word disambiguation.

C. Crowdsourcing

The burst of communication technologies has emerged virtual communities. People are now easily connected and can communicate, share and join together. Considering this new reality, industries and organizations discovered an innovative low-cost work force, which could save time and money in problem solving, as online recruitment of anonymous, also known as *crowdsourcing* [2], [13]–[15].

Since the seminal work of Surowiecki [16], the concept of *crowdsourcing* is expanding, mainly through the work of Jeff Howe [13], where the term crowdsourcing was definitely coined.

The underpinning idea behind crowdsourcing is that, under the right circumstances, groups can be remarkably intelligent and efficient. Groups do not need to be dominated by exceptionally intelligent people in order to be smart, and are often smarter than the smartest individual in them, that is the group decisions are usually better than the decisions of the brightest party.

As an example, if you ask a large enough group of diverse, independent people, to make a prediction or estimate a probability, and then average those estimates, the errors each of them makes in coming up with an answer will cancel themselves out. This means that virtually anyone has the potential to plug in valuable information [16], [17]. There are four conditions that characterize wise crowds [16]:

1) **Diversity of opinion**, as each person should have some private information, even if it is just an eccentric interpretation of the known facts.
2) **Independence**, related to the fact that people's opinion is not determined by the opinions of those around them.
3) **Decentralization**, in which people are able to specialize and draw on local knowledge.
4) **Aggregation**, related to the existing mechanisms for turning private judgments into a collective decision.

Besides the intelligent use of a group, there is another noteworthy advantage on using crowdsourcing, as there are tasks that are notoriously difficult for an algorithm to perform and quite simple for humans, like speech or image recognition language understanding, text summarization and labelling [18]. Taking advantage of these inherent capabilities, many crowdsourcing platforms emerged, such as the now widely used `Amazon Mechanical Turk` and `Yahoo! Answers`. However, the use of crowdsourcing face four key challenges: how to recruit contributors, what they can do, how to combine their contributions, and how to manage abuse [1].

Crowdsourcing capabilities in learning and classification strategies has been an emergent and bubbling area of research. In [19] crowdsourcing is used for the classification of emotion in speech, by rating contributors and defining associated bias. In [20] people that contribute to image classification are also rated to obtain cost-effective labels. Another interesting application is presented in [21], where facial recognition is carried out by requesting people to tag specific characteristics is facial images. In [22] crowdsourcing is used to process queries that neither database systems nor search engines can adequately answer, like ranking pictures by subject areas. Another application in data management research field is presented in [23], where Amazon's Mechanical Turk is used to write SQL-like queries to retrieve data that can not be achieved by a relational model. There are still few applications of crowdsourcing for text classification. In [24] economic news articles are classified using supervised learning and crowdsourcing.

III. Proposed approach

This section describes the proposed strategy to take advantage of using active learning and crowdsourcing to boost the performance of a baseline SVM in classification problems. In Fig. 1 we show the proposed framework.

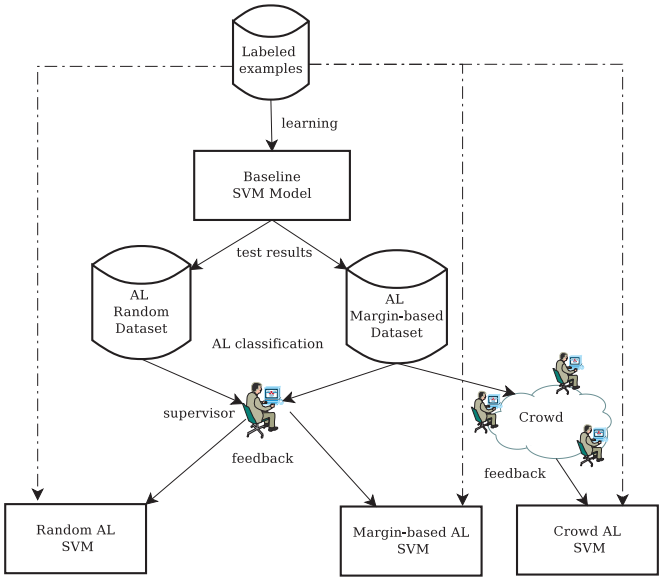

Figure 1. Proposed crowdsourcing active learning framework.

Our approach is threefold, resulting in the three final models in the figure:

- Random AL SVM: active learning with a random set of examples;
- Margin-based AL SVM: margin-based data set is created and it is correctly classified by a supervisor;
- Crowd AL SVM: the same margin-based data set is classified by crowdsourcing instead of the supervisor.

The main idea of using a random data set and a margin-based data set is to acquire the importance of the examples in the active learning process.

To define the margin-based data set a baseline SVM model is constructed and trained with labeled examples. As SVM classifies according to which side of the Optimal Separating Hyperplane (OSH) the examples fall, not all unlabeled points are classified with the same distance to the OSH. In fact, the farther from the OSH they lie, i.e. the larger the margin, more confidence can be put on their classification, since slight deviations of the OSH would not change their given class. Our criteria is to consider the examples where the SVM has less confidence. These examples, chosen criteriously, can make the SVM achieve greater accuracy, with the main idea of boosting the performance using fewer training labels.

The number of examples used can not be large, since the supervisor and the crowd will be asked to manually classify them. After being classified, they are integrated in the training set.

IV. Experimental Setup

In this section we start by describing humour classification and the Jester jokes data set, used in the experiments. We then proceed by detailing the pre-processing method and finally, we conclude by depicting the performance metrics used to evaluate the proposed approach.

A. Humour classification

Humour research in computer science has two main research areas: humour generation [25], [26] and humour recognition [8], [9], [27]. With respect to the latter, research done so far considers mostly humour in short sentences, like *one-liners*, that is jokes with only one line sentence, and the improvement of interaction between applications and users.

Humour classification is intrinsically subjective. Each one of us has its own perception of fun, hence automatic humour recognition is a difficult learning task that is gaining interest among the scientific community.

Classification methods used thus far are mainly text-based and include SVM classifiers, *naïve Bayes* and less commonly decision trees.

In [8] a humour recognition approach based in *one-liners* is presented. A data set was built grabbing *one-liners* from many websites with an algorithm and the help of web search engines. This humorous data set was then compared with non-humorous data sets like headlines from news articles published in the Reuters newswire and a collection of proverbs.

Another interesting approach [27] proposes to distinguish between an implicit funny comment and a not funny one. A 600,000 web comments data set was used, retrieved from the Slashdot news Web site. These web comments were tagged by users in four categories: funny, informative, insightful,

and negative, which split the data set in humorous and non-humorous comments.

B. Data set

The Jester data set contains 4.1 million continuous ratings (-10.00 to +10.00) of 100 jokes from 73,421 users and is available at: http://eigentaste.berkeley.edu. It was generated from Ken Goldberg's joke recommendation website, where users rate a core set of 10 jokes and receive recommendations from other jokes they could also like. As users can continue reading and rating and many of them end up rating all the 100 jokes, the data set is quite dense.

The data set is provided in three parts: the first one contains data from 24,983 users who have rated 36 or more jokes, the second one data from 23,500 users who have rated 36 or more jokes and the third one contains data from 24,938 users who have rated between 15 and 35 jokes. The experiments were carried out using the first part as it contains a significant number of users and rates for testing purposes, and for classification purposes was considered that a joke classified on average above 0.00 is a recommendable joke, and a joke below that value is non recommendable.

The jokes were split into two equal and disjoint sets: training and test. The data from the training set is used to select learning models, and the data from the testing set to evaluate performance.

C. Pre-processing methods

A joke is represented as the most common, simple and successful document representation, which is the vector space model, also known as *Bag of Words*. Each joke is indexed with the *bag* of the terms occurring in it, i.e., a vector with one component for each term occurring in the whole collection, having a value that takes into account the number of times the term occurred in the joke. It was also considered the simplest approach in the definition of term, as it was defined as any space-separated word.

Considering the proposed approach and the use of text-classification methods, pre-processing methods were applied in order to reduce feature space. These techniques, as the name reveals, reduce the size of the joke representation and prevent the mislead classification as some words, such as articles, prepositions and conjunctions, called *stopwords*, are non-informative words, and occur more frequently than informative ones. These words could also mislead correlations between jokes, so *stopword* removal technique was applied. *Stemming* method was also applied. This method consists in removing case and inflection information of a word, reducing it to the word stem. Stemming does not alter significantly the information included, but it does avoid feature expansion.

D. Performance metrics

In order to evaluate a binary decision task we first define a contingency matrix representing the possible outcomes of the classification, as shown in Table I.

	Class Positive	Class Negative
Assigned Positive	a	b
	(True Positives)	(False Positives)
Assigned Negative	c	d
	(False Negatives)	(True Negatives)

Table I
CONTINGENCY TABLE FOR BINARY CLASSIFICATION.

Several measures have been defined based on this contingency table, such as, error rate ($\frac{b+c}{a+b+c+d}$), recall ($R = \frac{a}{a+c}$), and precision ($P = \frac{a}{a+b}$), as well as combined measures, such as, the van Rijsbergen F_β measure [28], which combines recall and precision in a single score:

$$F_\beta = \frac{(\beta^2 + 1)P \times R}{\beta^2 P + R}. \quad (3)$$

F_β is one of the best suited measures for text classification used with $\beta = 1$, i.e. F_1, an harmonic average between precision and recall (4).

$$F_1 = \frac{2 \times P \times R}{P + R}. \quad (4)$$

V. EXPERIMENTAL RESULTS AND ANALYSIS

In this Section we manage to evaluate the performance obtained on processing Jester data sets against the three learning approaches described in Section III, namely random active learning SVM, margin-based active learning SVM and crowdsourcing SVM.

Averages were taken with 100 answers and processed just like described in the initial data set pre-processing methods (see Section IV-C). In order to evaluate the active learning strategies, we used 10 jokes, following the initial data set construction procedure IV-B. For the first experiment we carried out 30 runs, by randomly selecting 10 active examples. For the second experiment, the 10 active learning examples were chosen based on the confidence of the baseline SVM, thus aiming to consider those which would be more informative to the learning model. These jokes are then correctly classified and presented again as training examples to the SVM.

Finally, the same ten jokes criteriously chosen where presented to a crowd in order to be classified. The average values obtained where then used as training examples in order to improve classification performance.

Table II summarizes the performance results obtained. Analysing the table we can see that active learning is not a major condition for improving performance. When we

	Precision	Recall	F1
Baseline SVM	81.40%	92.11%	86.42%
Random AL SVM	84.36%	84.74%	83.81%
Margin AL SVM	87.80%	94.74%	91.14%
Crowd SVM	81.82%	94.74%	87.80%

Table II
COMPARATIVE RESULTS.

choose randomly the examples presented as active learning the performance can even decrease as can be seen by the presented values. However, when correctly chosen a remarkably important performance boost is noticed. Both recall, precision and F1 were improved by using this active learning approach, and the enhancements are robust regarding false positive and false negative examples.

When crowdsourcing is used we were able to verify a slight improvement considering baseline. Although, it is important to note that even being slight, this method also improves on all the metrics.

There are a few explanations for this minor improvement when compared with the achieved when using the margin-based active learning approach:

1) Humour is subjective, as it influenced by the contextual meaning of the joke, and can vary accordingly with culture, region, race or sex.
2) The definition of crowd is subjective, as it is difficult to acquire if the correct size of the crowd is used or that the crowd is diverse enough to provided substantial results.
3) The supervisor does not fail, although in some real-world problems it is impossible to use correct label data as active learning examples, others exist where the use of active learning is considered just to limit the number of examples given to a supervisor who can easily label correctly the presented data.
4) The crowd is not sure, as it is impossible to acquire if annotators can really classify the given examples. In this particularly example the crowd used was mostly Portuguese, i.e. non-English native, and some jokes were intrinsically related to the American culture, which could explain the results, yet just six jokes were correctly classified.

VI. CONCLUSIONS AND FUTURE WORK

In this paper we have presented a framework to evaluate active learning and crowdsourcing. Our aim was to evaluate the improvement of performance with the use of active learning methods when compared with baseline SVM. Two different active learning strategies were used. The first one used a supervisor that correctly classified the examples and the second approach used crowdsourcing to classify the same examples.

The results are presented in the Section V and reveal that an active learning strategy can outperform a baseline SVM model. The results obtained are very promising, as we are able to observe a major improve of all metrics.

Although it improved all metrics, it is noteworthy to refer that considering the specific problem we used to test our approach, humour classification, the improve of precision is remarkably important, specially in a humour recommendation application, where it is imperative to be assertive, but not so much crucial to retrieve every true positive. This last premise can be extended to most recommendation systems.

Regarding the use of crowdsourcing, and bearing in mind the successful accomplishments in many fields, we must state that crowdsourcing can be seen as a promising technique. However, the presented results show that it can be fallible in accomplishing specific tasks. Although it is important to note that we used an assertive supervisor, which can be unfeasible in many real-world problems.

It is also relevant to refer that our crowd can not be suited to this specific problem. Yet, it is somewhat difficult to evaluate the appropriateness of a crowd in such a subjective classification problem, and more research must be done in order to confirm its aptness. The evaluation of the crowd characteristics is for seen as future work.

REFERENCES

[1] A. Doan, R. Ramakrishnan, and A. Y. Halevy, "Crowdsourcing systems on the World-Wide Web," *Communications of the ACM*, vol. 54, no. 4, p. 86, apr 2011.

[2] D. C. Brabham, "Crowdsourcing as a Model for Problem Solving: An Introduction and Cases," *Convergence: The International Journal of Research into New Media Technologies*, vol. 14, no. 1, pp. 75–90, feb 2008.

[3] Y. Baram, R. El-Yaniv, and K. Luz, "Online choice of active learning algorithms," in *Proceedings of ICML-2003, 20th International Conference on Machine Learning*, 2003, pp. 19–26.

[4] V. Vapnik, *The Nature of Statistical Learning Theory*. Springer, 1999.

[5] T. Joachims, *Learning Text Classifiers with Support Vector Machines*. Kluwer Academic Publishers, Dordrecht, NL, 2002.

[6] S. Tong and D. Koller, "Support vector machine active learning with applications to text classification," *The Journal of Machine Learning Research*, vol. 2, pp. 45–66, 2002.

[7] M. Antunes, C. Silva, B. Ribeiro, and M. Correia, "A Hybrid AIS-SVM Ensemble Approach for Text Classification," *Adaptive and Natural Computing Algorithms*, pp. 342–352, 2011.

[8] R. Mihalcea and C. Strapparava, "Technologies That Make You Smile: Adding Humor to Text-Based Applications," *Intelligent Systems, IEEE*, vol. 21, no. 5, pp. 33–39, 2006.

[9] ——, "Making computers laugh: investigations in automatic humor recognition," in *Proceedings of the conference on Human Language Technology and Empirical Methods in Natural Language Processing*, ser. HLT '05. Stroudsburg, PA, USA: Association for Computational Linguistics, 2005, pp. 531–538.

[10] B. Settles, "Active learning literature survey," CS Technical Report 1648, University of Wisconsin-Madison, 2010.

[11] C. Silva and B. Ribeiro, "On text-based mining with active learning and background knowledge using svm," *Soft Computing - A Fusion of Foundations, Methodologies and Applications*, vol. 11, no. 6, pp. 519–530, 2007.

[12] A. K. McCallum and K. Nigam, "Employing EM and pool-based active learning for text classification," in *Proceedings of ICML-98, 15th International Conference on Machine Learning*. Morgan Kaufmann Publishers, San Francisco, US, 1998, pp. 350–358.

[13] J. Howe, "The Rise of Crowdsourcing," *Wired*, jun 2006.

[14] P.-Y. Hsueh, P. Melville, and V. Sindhwani, "Data Quality from Crowdsourcing: A Study of Annotation Selection Criteria," pp. 1–9, may 2009.

[15] O. Nov, O. Arazy, and D. Anderson, "Dusting for science: motivation and participation of digital citizen science volunteers," *Proceedings of the 2011 iConference*, pp. 68–74, 2011.

[16] J. Surowiecki, *The Wisdom of Crowds*. Doubleday, 2004.

[17] S. Greengard, "Following the crowd," *Communications of the ACM*, vol. 54, no. 2, p. 20, feb 2011.

[18] J. Barr and L. F. Cabrera, "AI gets a brain," *Queue*, vol. 4, no. 4, pp. 24–29, may 2006.

[19] A. Tarasov and S. Delany, "Using crowdsourcing for labelling emotional speech assets," in *ECAI - Prestigious Applications of Intelligent Systems*, 2010, pp. 1–11.

[20] P. Welinder and P. Perona, "Online crowdsourcing: rating annotators and obtaining cost-effective labels," in *IEEE Computer Society Conference on Computer Vision and Pattern Recognition workshops*, 2010, pp. 25–32.

[21] Y. Chen, W. Hsu, and H. Liao, "Learning facial attributes by crowdsourcing in social media," in *WWW 2011*, 2011, pp. 25–26.

[22] M. Franklin, D. Kossman, and T. Kraska, "CrowdDB: Answering queries with crowdsourcing," in *SIGMOD 2011*, 2011.

[23] A. Marcus, E. Wu, D. Karger, and S. Madden, "Crowdsourced databases: Query processing with people," in *CIDR 2011*, 2011.

[24] A. Brew, D. Greene, and P. Cunnigham, "The interaction between supervised learning and crowdsourcing," *NIPS 2010*, 2010.

[25] O. Stock and C. Strapparava, "Getting serious about the development of computational humor," in *IJCAI'03*, 2003, pp. 59–64.

[26] K. Binsted and G. Ritchie, "An implemented model of punning riddles," *arXiv.org*, vol. cmp-lg, jun 1994.

[27] A. Reyes, M. Potthast, P. Rosso, and B. Stein, "Evaluating Humor Features on Web Comments," in *Proceedings of the Seventh conference on International Language Resources and Evaluation (LREC'10)*, may 2010.

[28] C. van Rijsbergen, *Information Retrieval*. Butterworths Ed., 1979.