

---

# Using Crowdsourcing and Active Learning to Track Sentiment in Online Media

---

**Anthony Brew**

School of Computer Science & Informatics  
University College Dublin  
anthony.brew@ucd.ie

**Derek Greene**

School of Computer Science & Informatics  
University College Dublin  
derek.greene@ucd.ie

**Pádraig Cunningham**

School of Computer Science & Informatics  
University College Dublin  
padraig.cunningham@ucd.ie

## Abstract

Tracking sentiment in the popular media has long been of interest to media analysts and pundits. With the availability of news content via online syndicated feeds, it is now possible to automate some aspects of this process. There is also great potential to *crowdsource*<sup>1</sup> much of the annotation work that is required to train a machine learning system to perform sentiment scoring. We describe such a system for tracking economic sentiment in online media that has been deployed since August 2009. It uses annotations provided by a cohort of non-expert annotators to train a learning system to classify a large body of news items. We report on the design challenges addressed in managing the effort of the annotators and in making annotation an interesting experience.

## 1 INTRODUCTION

A recent article in the New York Times [18] discussed the emergence of a new business in sentiment analysis. The article reports on the emergence of companies that have begun to generate revenue streams by analyzing the reputation of their clients in online media, such as established news sources, blogs, and micro-blogs. The general problem of detecting and summarizing online opinion has also recently become an area of particular interest for researchers in the machine learning (ML) community [3].

In this paper we describe a demonstration application<sup>2</sup> that addresses some of the challenges in sentiment analysis of online content. The main technical innovation in this work is the use of annotations from a number of users to train a learning system to annotate a large number of news items. Rather than relying on polarity judgments from a single expert, such as an individual economist, the strategy adopted in this system is to generate trend statistics by collecting annotations from a number of non-expert users. These annotations are then used to train a classifier to automatically label a much larger set of news articles. It is worth emphasizing that the annotators are volunteers, so we are not dealing with crowdsourcing in the micro-task markets sense (*e.g.* Amazon's Mechanical Turk [9]), where annotators are paid for their efforts [8, 13]. The main reward for the annotators is the representation used in the annotation process itself – a *Really Simple Syndication* (RSS) feed

---

<sup>1</sup>Crowdsourcing is a term, sometimes associated with Web 2.0 technologies, that describes outsourcing of tasks to a large often anonymous community.

<sup>2</sup>See: <http://sentiment.ucd.ie>.

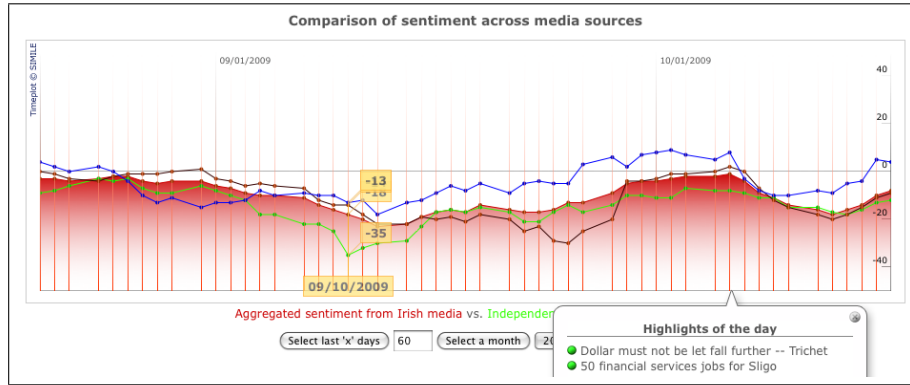


Figure 1: A screenshot of the time-plot generated by the system, which tracks economic sentiment from the various news sources over time.

providing a distillation of topical news stories. The system also helps *decompose* sentiment by providing tag clouds of discriminating positive and negative terms (see Figure 3), along with lists of highly positive and negative articles (see website).

The combination of active learning and crowdsourcing has a number of advantages in the context of sentiment analysis:

- Using a classifier, a large number of unlabeled items can be classified to provide robust statistics regarding sentiment trends.
- Statistics can be generated after the annotation process ends. The extent to which this can be done depends on the amount of *concept drift* that occurs over time in the specific domain of interest.
- The article selection process ensures a diverse annotation load that provides the annotator with a good overview of the day’s news.

In this paper we describe the overall architecture of the system (see Figure 2) and present some of the challenges addressed in making best use of the annotators efforts and in making the annotation a rewarding exercise. In particular we discuss the related problems of *consensus* and *coverage* in collecting annotations.

Given that the main objective of the system is to generate plots of the type shown in Figure 1, it is important that the classifier should not be biased. In other work [4] we have shown that nearest neighbor, naïve Bayes, and Support Vector Machine (SVM) classifiers are biased toward the majority class in our task. We have presented a strategy for managing this bias in the training data. This research is not reported here for space reasons, however the details are available in [4].

The remainder of the paper is structured as follows. In the next section we provide an overview of research related to our task. In Section 3 we describe the system in more detail, and outline our strategy for integrating crowdsourcing and supervised learning. Further detail on the approach for selecting articles for annotation is given in Section 4. In Section 5 the trade-off between annotation consensus and coverage is discussed. The paper finishes with some conclusions on how this system might be applied to other sentiment analysis tasks.

## 2 RELATED WORK

The general problem of detecting the polarity (positive or negative) of opinions in online content has recently become an area of particular interest for researchers in the natural language processing and machine learning communities. Common approaches have included the identification of authors’ attitudes based on applying standard text classification techniques to document bag-of-words representations [11], searching for opinion-carrying terms in documents [1], and frequent pattern mining to identify syntactic relations between sequences of terms that may be indicative of sentiment po-

larity [10]. Most frequently these techniques have been applied to tasks such as classifying movie reviews [11] or product reviews [3] based on the polarity of review text.

Traditionally, datasets for sentiment analysis tasks have been manually constructed by small groups of expert annotators with specific training (*e.g.* the MPQA corpus [17]). While this approach to annotating sentiment in text corpora can provide detailed, high-quality data, it will often be infeasible in real-world tasks due to time constraints or lack of access to domain experts. As an alternative, services such as Mechanical Turk [9] have demonstrated the utility of harnessing crowds of non-expert users to perform time-consuming labeling tasks. There is already a significant research literature on the problem of aggregating a number of medium quality annotations in order to generate a good quality annotation. Two important early contributions in this area are the work of Dawid and Skene [6] and the work of Smyth *et al.* [15]. Recently there has been renewed interest in this area with the advent of crowdsourcing as a fast and effective mechanism of generating medium quality annotations [16, 8, 7, 13]. A key question in this area relates to the importance placed on data quality. Snow *et al.* show that, for text annotation tasks similar to that addressed in our work, crowdsourced annotators are not as effective individually as experts. But when non-expert opinions are aggregated together, it is possible to produce high-quality annotations [16]. So this work establishes the merit of aggregating a number of annotations in order to generate good quality annotations.

The question of the balance between data coverage and annotation quality arises frequently in the literature. Raykar *et al.* [13] proposed a strategy that simultaneously induces “ground truth” (or gold standard) from multiple annotations, while also building a classifier based on this labeling. The authors suggest that having effective annotators is more important than data coverage, and emphasize the use of multiple annotations for each item, in conjunction with weights for annotators based on their agreement with the induced ground truth. Smyth *et al.* [15] also highlighted the difficulty of performance evaluation in tasks where annotations are available from multiple annotators, but no ground truth is available as a reference. In such cases we must rely on annotator consensus as a proxy when measuring annotation quality.

### 3 SYSTEM DESCRIPTION

The primary objective of our system is to produce unbiased assessments of sentiment in a dynamic collection of news articles, so that trends and differences between sources can be identified and visualized as shown in Figure 1. In the system implementation, articles are collected from a pre-defined set of RSS feed URLs published by the news sources of interest. After applying a relevance classifier, most articles not pertaining to economic news are filtered from the candidate set. From the remaining relevant articles, a subset is chosen based on an appropriate article selection mechanism. The resulting subset of articles is then presented via an RSS feed to the annotators, who are encouraged to label the articles as *positive*, *negative*, or *irrelevant*. These annotations are subsequently used to retrain the classification algorithms on a daily basis.

The main components of the system are outlined in Figure 2. The selection of articles for annotation takes place at (A), and the polarity classification and bias correction happens at (B). Given that there is a large collection of articles to be annotated (either manually or by the classifier), the article selection policy for manual annotation has a considerable impact on the overall annotation quality. This issue is discussed in detail in Section 4, while a solution for bias correction is proposed in [4].

#### 3.1 The Annotation Process

Articles are collected from a pre-defined set of RSS feed URLs at the beginning of each day. In cases where only short descriptions are provided for RSS items, the original article body text is retrieved from the associated item URL. Those articles coming from the same domain (*i.e.* from the same news source) are grouped together. After applying the relevance classifier as described previously, articles not pertaining to economic news are filtered from the candidate set. From the remaining relevant articles, a diverse subset of approximately ten articles is chosen using the article selection mechanism (see Section 4). The resulting subset of articles is then published as a customized RSS feed for each of the system’s users.

To support the annotation process, a footer is appended to each RSS item in the custom feed containing links corresponding to the three annotation choices: *positive*, *negative*, or *irrelevant*. Selecting a

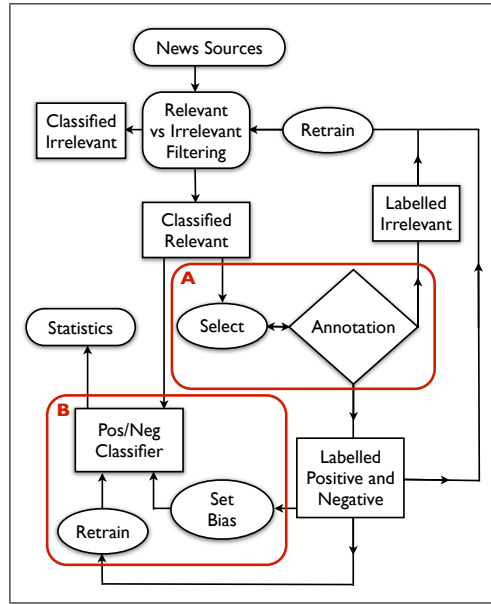


Figure 2: Overall design of the economic sentiment analysis system. The important components are (A) the article selection and annotation process, and (B) the training of the classifier where classification bias is controlled.

link submits a single vote to the system on the article in question. The use of an RSS feed as a means of both delivering articles to be annotated and receiving annotation votes is designed to minimize the work-load of the annotation procedure in the context of a user’s existing routine. We found that many users integrated the process as part of their existing news-reading habits – either via an online RSS reader (*e.g.* Google Reader) or a desktop news aggregator (*e.g.* Apple Mail). For those users who do not currently make use of an online or desktop RSS reader, many modern web browsers include the facility to render and display RSS feeds as web pages.

Annotations received from users are subsequently used to retrain the classification algorithms on a daily basis. The effectiveness of the next day’s relevance filtering process is improved based on newly-collected *relevant* (*i.e.* *positive* or *negative*) or *irrelevant* votes. Similarly, articles that have been annotated as either *positive* or *negative* are included when re-training the second classifier. This is used to improve the quality of the summary statistics and visualizations on the web interface, which we describe in the next section.

### 3.2 Web Interface

In a system such as this the value for users is based on a variety of channels with which to access relevant content, many of which are enabled by the classification components. For example, the statistical visualizations of Figure 1 reward users with a sense of how their efforts are contributing to the system as a whole, as well as providing direct access to trending sentiment with current news. Users can review lists of the most positive, negative, and controversial articles for instance. Yet another example is presented in Figure 3, where users can benefit from tag-cloud summaries which highlight the most representative terms that appear in the positive or negative articles around a selected date.

### 3.3 Evaluation Data

While the system is in continuous operation, the evaluation presented here covers articles retrieved from three online news sources (RTE, The Irish Times, The Irish Independent) using the system outlined in Figure 2 during a three month period (July to October 2009). A subset of these were annotated on a daily basis by a group of 33 volunteer users. The first month constituted a “warm-up” period, which allowed us to train the relevance classifier to a point where it achieves approximately



Figure 3: A screenshot of a tag cloud generated by the system, highlighting terms associated with *negative* sentiment.

90% accuracy. This provided an initial dataset containing 3858 articles, with 2693 user annotations covering 354 individual articles. After this warm-up period the data from August onward was the main focus of the evaluation. This second “main” dataset comprises 12469 documents, with 6910 user annotations resulting in 1306 labeled articles. Both datasets have been made available online<sup>3</sup> for further research.

### 3.4 Baseline Classification

For the classification components of the system, we considered three supervised learning techniques that have previously been effective in text classification tasks [5]. These are naïve Bayes, SVMs, and  $k$ -nearest neighbor ( $k$ -NN). In order to select the classifier that was best suited to our task, we performed a baseline assessment using cross-validation. In all cases we follow Pang *et al.* [11] who suggested the use of unigram bag-of-words features to represent documents, although we do make use of term frequency information rather than merely looking at the presence or absence of terms.

The results of the evaluation are shown in Table 1. Accuracy figures are reported for each of the three classification techniques on two different tasks (positive vs. negative and relevant vs. irrelevant). We also report AUC (area under the ROC curve) figures [12], as these consider classifier performance across a range of thresholds and are thus independent of bias considerations.

Measure	<i>Positive vs. Negative</i>			<i>Relevant vs Irrelevant</i>		
	Bayes	SVM	$k$ -NN	Bayes	SVM	$k$ -NN
AUC	0.80	0.82	0.71	0.90	0.88	0.68
Accuracy	75%	77%	72%	85%	81%	76%

Table 1: Baseline accuracies for the three classifier types on the two classification tasks.

These results corroborate the previous findings in [11], which showed that SVMs tend to only marginally out-perform naïve Bayes in sentiment classification tasks. The  $k$ -NN classifier did not perform well in the evaluation and was not considered further. The Bayes classifier performed best on the relevance task and was competitive on the positive vs. negative task. In addition, since many of our experiments here involve active learning-style scenarios, algorithm time complexity is an important consideration. In this respect the linear training time of naïve Bayes is preferable to the cubic training time of SVMs. Another important consideration is the fact that the Bayes classifier is easier to update than the SVM because the SVM is sensitive to parameter selection. For these reasons we employed a naïve Bayes classifier in our sentiment analysis system.

<sup>3</sup>See <http://mlg.ucd.ie/sentiment>

## 4 ARTICLE SELECTION

As described previously, only a fraction of all articles retrieved from the news sources will be presented to the users for manual annotation. A natural question arises as to how an appropriate subset of articles should be chosen on a given day – this corresponds to component (A) in Figure 2. In some respects this problem resembles the task of query selection in active learning, where the goal is to select the most informative unlabeled items to present to the oracle. However, another goal to consider in the context of crowdsourced annotation is the selection of a diverse set of items that will be of interest for the annotator. We wish to incentivize users to annotate articles by providing them with a useful summary of the day’s economic news stories, delivered in the form of an RSS feed. For this reason we wish to avoid duplicate or highly-similar articles.

To identify a diverse set of articles that provides a representative summary of the day’s economic news, we apply a clustering-based article selection strategy. Firstly we construct  $k$  clusters of articles by merging all pairs of articles with cosine similarity above a threshold  $\tau \in [0, 1]$ . This is equivalent to applying complete-linkage agglomerative clustering and choosing a merging cut-off threshold  $\tau$ . From the set of clusters, we then choose a subset  $k' < k$  using weighted farthest-first traversal [2]. This leads to the selection of a sufficiently diverse set of clusters, while also ensuring that large clusters (*i.e.* representing dominant news stories for a particular day) are likely to be selected. The most representative article from each of the  $k'$  clusters (*i.e.* most similar to the cluster centroid) is then selected for annotation. In practice we found that approximately  $k' = 10$  articles was a reasonable number of articles to present to users for annotation each day.

To examine the utility of the proposed selection strategy, we compare the strategy for different values of  $\tau$  with a baseline random selection strategy. While it is not possible to evaluate the strategy on a daily basis since we only have  $\approx 10$  labeled articles per day, we can approximate the daily selection process by applying article selection to a set of labeled articles from a window of 7 consecutive days. Figure 4 shows the AUC performance of the clustering strategy ( $\tau \in [0.3, 0.5]$ ) as 10 additional articles are selected from each 7 day window and added to the classifier’s training set. The clustering strategy out-performs random selection for all  $\tau$  parameter values tested, with the best AUC scores achieved by  $\tau = 0.5$  which corresponds to a more conservative agglomeration of articles.

We observe that, when values of  $\tau \geq 0.6$  are used, many singleton clusters are produced, even in cases where articles cover the same news story. In practice a selection of a conservative threshold  $\tau \approx 0.5$  for our task has the effect that articles on distinct stories are not treated as being identical, while highly-similar articles, reporting on the same news story, are grouped together so that only the most representative article is presented to the annotators. This ensures that the article selection strategy is not only beneficial for the subsequent training phase in terms of covering as much of the

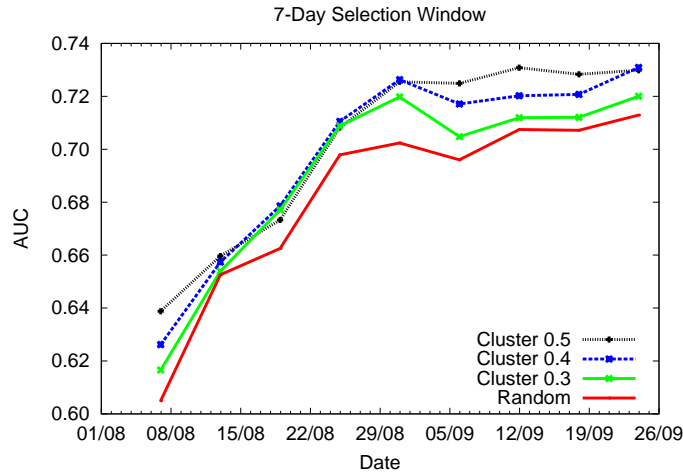


Figure 4: Comparison of article selection using a clustering strategy versus random selection (averaged over multiple runs).

domain as possible (discussed further in the next section), but also ensures the selection of a diverse set of reading material for the annotators.

## 5 CONSENSUS VERSUS COVERAGE

The development of crowdsourcing has provided a fast and effective mechanism for obtaining annotations [8, 7, 13, 16]. When such non-expert opinions are combined together, it may be possible to produce higher quality aggregated annotations [16]. While this facility is relevant to our task, there is one important difference. In our task good quality annotations are not an end in themselves. Rather we require a body of annotated data that can be used to subsequently train a classifier. Given the role of the classifier in the overall sentiment analysis system, a natural question arises – is it better to use the annotation “budget” to produce consensus judgments, or should annotator effort be spread out across as many items as possible to provide better coverage of the domain? Which is preferable – 300 single annotations on 300 items, or 60 annotations based on 5 annotations per item?

In the remainder of this section we develop a policy for obtaining multiple annotations for articles, which considers the trade-off between these two important considerations:

- **Consensus:** How does the agreement between the annotators, or lack thereof, affect the performance of the classifier?
- **Coverage:** With a limited annotation budget, how can we make effective use of this budget to adequately cover the domain?

### 5.1 Consensus

The notion of annotation quality needs to be treated with care when measuring sentiment. In some tasks, aggregated high-quality annotations will always correspond to the “correct answer”. Whereas in the context of the Irish economy dataset, the answer is likely to be far more subjective. Indeed expert economists or political scientists might have strongly divergent opinions regarding the topics discussed in many of the news articles.

At this point it is worth formally defining *consensus* – it represents the margin by which users agree on the polarity of an article. A simple quantitative measure for the degree of consensus on a single article is given by:

$$\text{consensus} = \frac{|\text{votes positive} - \text{votes negative}|}{|\text{votes positive} + \text{votes negative}|}$$

While there exists strong agreement between annotators on many articles, a significant proportion of articles (45%) do not achieve 100% consensus, see the report by XXX *et al.* for details [4].

To measure the impact of consensus on the classifier used in our system, we selected a set of 350 articles that had five or more annotations. Articles were then labeled according to their majority vote, and then separated into positive and negative sets. Note that the 350 articles were selected to ensure these sets were balanced in size. Bayes classifiers were trained by presenting this data using two different ordering policies: presenting the articles from low consensus to high consensus (“Weak to Strong”), and vice versa (“Strong to Weak”). At each step, an article was added from both the positive and negative sets to ensure the classifier remained balanced. This experiment was run in a 10 fold cross validation setup and repeated 100 times with random shuffling to eliminate any effects arising from data ordering.

The results from this experiment, in terms of AUC scores, are shown in Figure 5. It is clear that articles with a high level of consensus are very beneficial for training the classifier, while articles on which consensus is low are far less useful. Indeed there is some evidence that the learning process would be better off without them.

### 5.2 Coverage

Given the trade-off between coverage and consensus we examined the impact of different budgeting policies. We evaluate three alternatives, single annotation votes, best of three, and best of five votes.

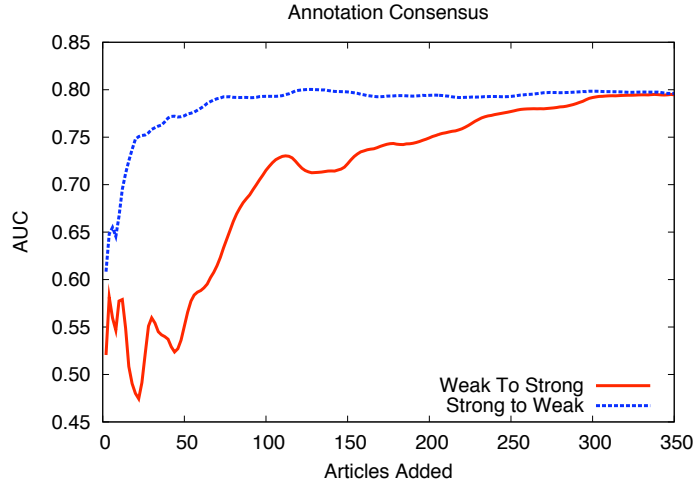


Figure 5: Learning curves where articles were presented using two different ordering policies. In the “Weak to Strong” policy, articles with lower consensus were added first, while in “Strong to Weak”, articles with higher consensus were added first.

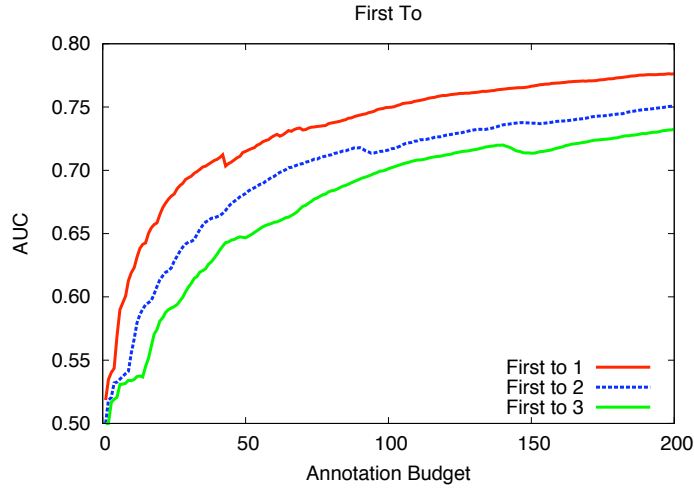


Figure 6: This graph shows how the learning curve improves based on different strategies for spending the annotation budget. The higher coverage afforded by the “First to 1” strategy proves most effective.

To avoid unnecessary spending of the budget, additional votes are not sought for an article if a clear majority has already been achieved. For this reason the alternative budgeting strategies are referred to as “First to 1”, “First to 2” and “First to 3” in Figure 6. This experiment entailed the same 100 times 10-fold cross validation setup as before.

The results in Figure 6 suggest that coverage is more important than consensus on this annotation task. Labeling a large volume of articles here proved more important than obtaining multiple votes on individual articles. It is worth emphasizing that this is the situation when learning performance is graphed against annotation effort. If we plot performance against the *number* of annotated articles (ignoring annotation cost) the consensus annotations do best.

In the next section we show that coverage in the training data is not always more important than consensus. The balance will depend crucially on the level of disagreement between the users participating in the annotation system. We also show that all annotators do not contribute equally to the system – some are more useful than others.



### 5.3 Consensus and Coverage

In the evaluation performed in the previous section, we observed that coverage was particularly effective in improving classification performance because in 89% of the cases, individual annotators agreed with the majority. To examine what happens in situations where the consensus is lower, we followed the approach described in [14] by adding noise to the training data. To do this 25% of the annotations were flipped in order to decrease annotator-majority agreement to approximately 67%. When the evaluation shown in Figure 6 is repeated with this data the results are quite different, as shown in Figure 7. Early in the learning process we see that coverage is still important as the classifier benefits from a breadth of examples. However, as learning continues, the availability of more reliable consensus labels assigned to articles by requesting a 2<sup>nd</sup> and 3<sup>rd</sup> opinion becomes more important. In contrast, the simple strategy of attaining maximum coverage begins to become less competitive. The evaluations shown in Figure 6 and Figure 7 demonstrate that the management of the annotation budget will often depend on the level of agreement among annotators. In our specific annotation task, the average level of inter-annotator agreement is high, and therefore it is less important to spend annotation effort to obtain consensus opinions.

The final issues we consider in this section concerns the variation between annotators. From the evaluation presented in Section 5.2, one might conclude that having just one user annotate all articles would be sufficient. However, when we look at the extent to which individual users agree with the consensus label, the level of agreement ranges from around 75% to 95%. Donmez *et al.* [7] recognized the importance of using strong annotators, and described a strategy for identifying weaker annotators. These weaker annotators were then removed from the annotation process in order to make best use of the annotation budget. To demonstrate the need for care when selecting annotators, we repeat the experiment shown in Section 5.2 for the single annotation strategy. However, rather than randomly selecting annotations for each article from those available in the data, we select annotations from the “strongest” annotator (*i.e.* user having the highest agreement with the consensus opinion) and “weakest” annotator (*i.e.* user having the lowest agreement with the consensus opinion).

The difference in AUC performance shown in Figure 8 clearly highlights the benefit of using strong annotators. It is interesting to note that the best single annotator is almost as good as using the consensus judgment (referred to as the “Oracle”) to train the system. These results motivate an effective way of managing the annotation budget. Once annotators that are close to the consensus opinion have been identified, other less informative annotators can be dropped from the process with little or no deterioration in classifier performance.

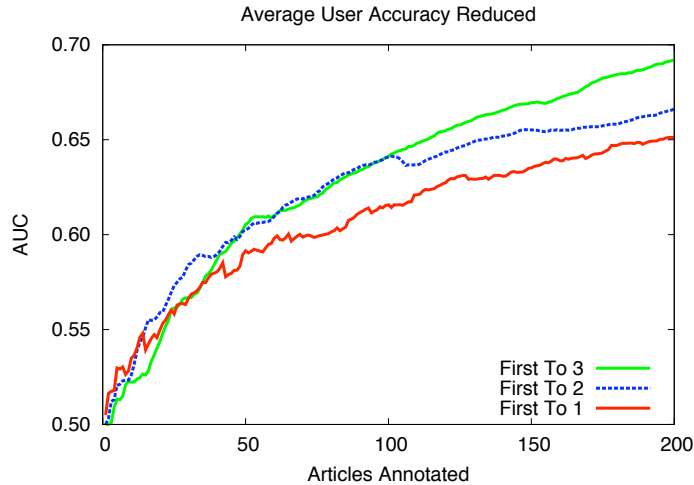


Figure 7: When reducing average annotator-majority agreement to 67%, coverage is still important in the early stages of the learning curve. By acquiring more annotations per article, consensus can be attained and, if a sufficient budget is available, classification performance can be improved.

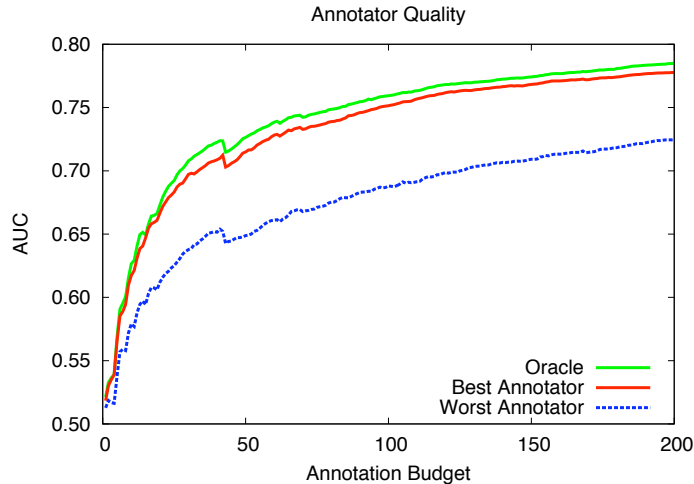


Figure 8: This figure demonstrates the benefits of strong annotators over weak annotators. Training with annotations that use the strongest annotator only is almost as effective as training on the majority consensus annotations – the “Oracle” in this graph.

## 6 Conclusions

We have presented an analysis of the challenges in training a sentiment analysis system using data collected from non-expert annotators. The objective of the system is to produce unbiased aggregate statistics on sentiment for large collections of news articles. In this paper we have discussed the benefits of this strategy that combines crowdsourcing and machine learning and we have focused on the specific challenge of managing the trade-off between coverage and consensus in the annotation process. We have also considered the related issue of selecting a diverse set of items for annotation in order to extend coverage and improve the annotation experience. We have shown elsewhere how to manage bias in the training process [4] so that the system will continue to produce accurate trend statistics such as that given in Figure 1 for at least two months after manual annotation is stopped.

Our main conclusion is that the commissioning of a system such that described here should be preceded by a data *characterization* phase. This would explore the extent of the agreement between annotators and the amount of skew in the data. Our first important finding is that, if there is good agreement between annotators, then annotation effort should be expended on maximizing coverage rather than on identifying consensus. Our second finding is that, even when the skew in the data is modest, there is a clear need to correct for bias in the training of the classifier.

## Acknowledgments

This research was supported by Science Foundation Ireland (SFI) Grant Nos. 05/IN.1/I24 and 08/SRC/I1407.

## References

- [1] G. Attardi and M. Simi. Blog mining through opinionated words. In *Proc. 15th Text REtrieval Conference (TREC 2006)*, 2006.
- [2] S. Basu, M. Bilenko, and R. J. Mooney. A probabilistic framework for semi-supervised clustering. In *Proc. 10th ACM SIGKDD International Conference on Knowledge Discovery and Data mining (KDD’04)*, pages 59–68, 2004.
- [3] J. Blitzer, M. Dredze, and F. Pereira. Biographies, bollywood, boom-boxes and blenders: Domain adaptation for sentiment classification. In *Proc. 45th Annual Meeting of the Association of Computational Linguistics*, pages 440–447, 2007.

- [4] A. Brew, D. Greene, and P. Cunningham. Is it Over Yet? Learning to Recognize Good News in Financial Media. Technical Report UCD-CSI-2010-1, University College Dublin, January 2010.
- [5] P. Cunningham, M. Cord, and S. Delany. Supervised Learning. In M. Cord and P. Cunningham, editors, *Machine Learning Techniques for Multimedia: Case Studies on Organization and Retrieval*, pages 21–49. Springer, 2008.
- [6] A. Dawid and A. Skene. Maximum likelihood estimation of observer error-rates using the EM algorithm. *Journal of the Royal Statistical Society. Series C (Applied Statistics)*, 28(1):20–28, 1979.
- [7] P. Donmez, J. G. Carbonell, and J. Schneider. Efficiently learning the accuracy of labeling sources for selective sampling. In *Proc. 15th ACM SIGKDD International Conference on Knowledge Discovery and Data mining (KDD '09)*, pages 259–268, 2009.
- [8] P. Hsueh, P. Melville, and V. Sindhvani. Data Quality from Crowdsourcing: A Study of Annotation Selection Criteria. In *Proc. NAACL HLT 2009 Workshop on Active Learning for Natural Language Processing*, pages 27–35, 2009.
- [9] A. Kittur, E. H. Chi, and B. Suh. Crowdsourcing user studies with mechanical turk. In *Proc. 26th Annual ACM Conference on Human Factors in Computing Systems (CHI'08)*, pages 453–456, 2008.
- [10] S. Matsumoto, H. Takamura, and M. Okumura. Sentiment classification using word sub-sequences and dependency sub-trees. In *Proc. 9th Pacific-Asia Conference on Knowledge Discovery and Data Mining (PAKDD'05)*, pages 301–310. Springer, 2005.
- [11] B. Pang, L. Lee, and S. Vaithyanathan. Thumbs up?: Sentiment classification using machine learning techniques. In *Proceedings of the ACL-02 conference on Empirical methods in natural language processing*, volume 10, pages 79–86, 2002.
- [12] F. Provost, T. Fawcett, and R. Kohavi. The case against accuracy estimation for comparing induction algorithms. In *Proc. 15th International Conference on Machine Learning (ICML'98)*, pages 445–453, 1998.
- [13] V. Raykar, S. Yu, L. Zhao, A. Jerebko, C. Florin, G. Valadez, L. Bogoni, and L. Moy. Supervised Learning from Multiple Experts: Whom to trust when everyone lies a bit. In *Proc. 26th Annual International Conference on Machine Learning (ICML'09)*, pages 889–896, 2009.
- [14] V. Sheng, F. Provost, and P. Ipeirotis. Get another label? improving data quality and data mining using multiple, noisy labelers. In *Proceeding of the 14th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 614–622, 2008.
- [15] P. Smyth, U. Fayyad, M. Burl, P. Perona, and P. Baldi. Inferring ground truth from subjective labelling of Venus images. *Advances in Neural Information Processing Systems (NIPS'95)*, 1995.
- [16] R. Snow, B. O'Connor, D. Jurafsky, and A. Y. Ng. Cheap and fast—but is it good?: evaluating non-expert annotations for natural language tasks. In *Proc. Conference on Empirical Methods in Natural Language Processing (EMNLP'08)*, pages 254–263. Association for Computational Linguistics, 2008.
- [17] T. Wilson, J. Wiebe, and P. Hoffmann. Recognizing contextual polarity in phrase-level sentiment analysis. In *Proceedings of the conference on Human Language Technology and Empirical Methods in Natural Language Processing*, pages 347–354. Association for Computational Linguistics, 2005.
- [18] A. Wright. Mining the web for feelings, not facts. *The New York Times*, 24 August 2009.