Sentiment Analysis of Online Media

Salter-Townshend, Michael and Murphy, Thomas Brendan
University College Dublin, School of Mathematical Sciences
Belfield, Dublin 4
Ireland
E-mail: michael.salter-townshend@ucd.ie brendan.murphy@ucd.ie

We present a joint estimator for observer bias and term classification in the context of media sentiment analysis. We analyse an Irish media dataset comprising user annotations of online news articles as having negative, positive or irrelevant impact. We include estimates for user bias in the analysis; rather than a straightforward majority vote determining the true sentiment, we compute an estimated sentiment calculated on a weighted sum of the user supplied annotations. These weights are the user biases and we estimate the user bias matrices using an Expectation-Maximisation algorithm which includes finding the expected but unobserved true sentiment in the articles. A classifier such as Naive-Bayes may then be trained on the words in the articles using these expected sentiments. Joint estimation of both the user biases and the classifier parameters is demonstrated to be superior to estimation of the bias followed by the estimation of the classifier parameters.

Introduction

Our work is in the context of sentiment analysis of news articles but our methods are readily applicable to any classification task in which the classifier is trained on (potentially) biased annotations. Our goal is to increase the accuracy of both the annotation based labelling and the classifier.

Many existing classifiers do not take into account user (annotator) bias in reporting; a simple majority vote is used to select article type from the observed annotations. Most recently [1] and [2] propose methods to address the problem of multiple imperfect annotations and classification. [1] deals with the labelling of clinical reports and uses Bayesian models with Gaussian processes for classification and ordinal regression. [2] address the problem of training a classifier with multiple imperfect annotations by extending the model of [3] to learn a classifier at the same time as the annotator biases via maximum likelihood; this work is similar to the approach developed herein. Specifically, they train a logisitic regression classifier and learn the sensitivity and specificity of the annotators in the context of binary labelling. The model that we present differs from that paper in that we explore a trinary labelling system (an arbitrary finite number of categories is possible) and train a Naïve-Bayes classifier. The contribution of our work is to demonstrate the method with another classifier, a greater number of potential labels and to report upon the comparative effectiveness of our approach on the online media sentiment dataset.

We validate the joint estimator first on a simulated dataset, for which the ground truth of both the annotator biases and the true types are known. We can thus calculate performance scores for both the decoupled estimation method (learn the biases and then train the classifier) and the joint estimation model. We demonstrate the superiority of the joint estimator for various levels of bias and then apply it to the media-sentiment dataset.

Sentiment Data

The Irish media sentiment dataset that we analyze is a subset of the data described in detail in [4] and [5]. The dataset is comprised of 1049 articles collected from 3 online Irish news services (rte.ie, irishtimes.com and independent.ie), collected fro July to October 2009. Thirty one volunteers have annotated an average of 834 of these articles as having either negative, positive or irrelevant impact on the Irish economy at time of press. There are $70,873$ word terms appearing in these articles. In order to reduce the impact of words that are too common (such as at, the, and, etc) we eliminate words that appear in more than 1000 articles. We also eliminate words that appeared in less than 30 articles. To further reduce the dimensionality of the data, we selected the top 300 most negative words (as indicated by a simple majority vote classifier), the top 300 most positive words and the top 300 most irrelevant words only.

The data, along with a description of the collection mechanism are described in detail in [4]; they note that 45% of the articles do not have consensus annotations and that "there is some evidence that the learning process would be better off without them [articles with low consensus]". The authors of that paper examined k nearest neighbours and SVM classifiers also and settled on Naïve-Bayes following cross-validation.

Model

We model the annotator bias as per [3]. Error rates, or biases in reporting, are modelled via a matrix of conditional probabilities for each annotator. i.e. the probability that annotator k records annotation j given a *true* (but unobserved) type i is denoted by $\pi_{ij}^{(k)}$. These probabilities necessarily sum to unity across j for each i and k. The observed annotations are thus a probabilistic (in this case multinomial) function of these π matrices. Hence, the likelihood for the recorded annotations y on article a given a true type i by all annotators k in $1, \ldots, K$ is given by

$$(1) \qquad \mathcal{L}(\pi | y_a^{(1)}, y_a^{(2)}, \ldots, y_a^{(K)}, i) \propto \prod_k^K \prod_j^J (\pi_{ij}^{(k)})^{y_{aj}^{(k)}}$$

where J is three in our sentiment levels (negative, positive and irrelevant).

We now introduce the true types; let the true type of article a be T_a, where $T_{ai} = 1$ if the article is of type i and $T_{ai} = 0$ otherwise. Hence, the likelihood of the full dataset (including unobserved true types) across all A articles is

$$(2) \qquad \mathcal{L}(\pi, p | y^{(1)}, y^{(2)}, \ldots, y^{(K)}, T_1, T_2, \ldots, T_A) \propto \prod_a^A \prod_i^J \left\{ p_i \prod_k^K \prod_j^J (\pi_{ij}^{(k)})^{y_{aj}^{(k)}} \right\}^{T_{ai}}$$

where p_i is the marginal probability of type i.

Another goal of the sentiment analysis described in [4] is to train a classifier to distinguish which word terms appear in which types of article (negative, positive or irrelevant to the Irish economy). The trained classifier may then be used to automatically label un-annotated articles. Although word-term frequencies are available in the dataset, we model only the presence or absence of these features (word terms). We therefore employ a Bernoulli likelihood for term w_n appearing in article a given it is of type i. That is, we use a Naïve-Bayes classifier to learn the probability of an article type given the words that appear in the article.

The product of Bernoullis likelihood for all N word terms w_a appearing in article a given that it is of type i is then

$$(3) \qquad \mathcal{L}(\theta | T_a, w_a) = \prod_i^J \left\{ \prod_n^N (\theta_{ni})^{w_{an}} (1 - \theta_{ni})^{1 - w_{an}} \right\}^{T_{ai}}$$

where θ_{ni} is the probability that word term w_n appears in an article of type i.

The full likelihood for the data is then a product of Equation (2) and a term in the form of Equation (3) for each article.

Algorithm

Since T, p and π are unknown in Equation (2), we proceed as per [3]; we first estimate initial values of the missing data T. We then extend the EM algorithm used in [3] to yield a joint estimation that learns θ within the same EM iteration loop as it learns the values of missing data (true types T), the marginal probabilities p and annotator bias matrices π. The algorithm proceeds as follows:

```
1. for all articles $a$:
2.     initialize $T$ using $\hat{T}_{ai} = \mathbf{E}[T_{ai}] = \sum_k y_{ai}^{(k)}/K$
3.     initialize $p$ using $\hat{p}_i = \sum_a T_{ai}/A$
4.     estimate all $\pi$ values via maximum likelihood expression
   (4)     $\hat{\pi}_{ij}^{(k)} = \frac{\sum_a \hat{T}_{ai} y_{aj}^{(k)}}{\sum_j \sum_a \hat{T}_{ai} y_{aj}^{(k)}}.$
5.     estimate all $\theta$ and $p$ via maximum likelihood expressions
   (5)     $\hat{\theta}_{ni} = \frac{\sum_a w_{an} \hat{T}_{ai}}{\sum_a \hat{T}_{ai}}$ and $\hat{p}_j = \frac{\sum_a \hat{T}_{aj}}{A}.$
6.     re-estimate $T$ using
   (6)     $\hat{T}_{ai} = \mathbf{E}[T_{ai}] = \frac{p_i \prod_k^K \prod_j^J (\hat{\pi}_{ij}^{(k)})^{y_{aj}^{(k)}} \prod_n^N (\hat{\theta}_{ni})^{w_{an}} (1-\hat{\theta}_{ni})^{1-w_{an}}}{\sum_i p_i \prod_k^K \prod_j^J (\hat{\pi}_{ij}^{(k)})^{y_{aj}^{(k)}} \prod_n^N (\hat{\theta}_{ni})^{w_{an}} (1-\hat{\theta}_{ni})^{1-w_{an}}}.$
7. repeat 4 to 6 until convergence
```

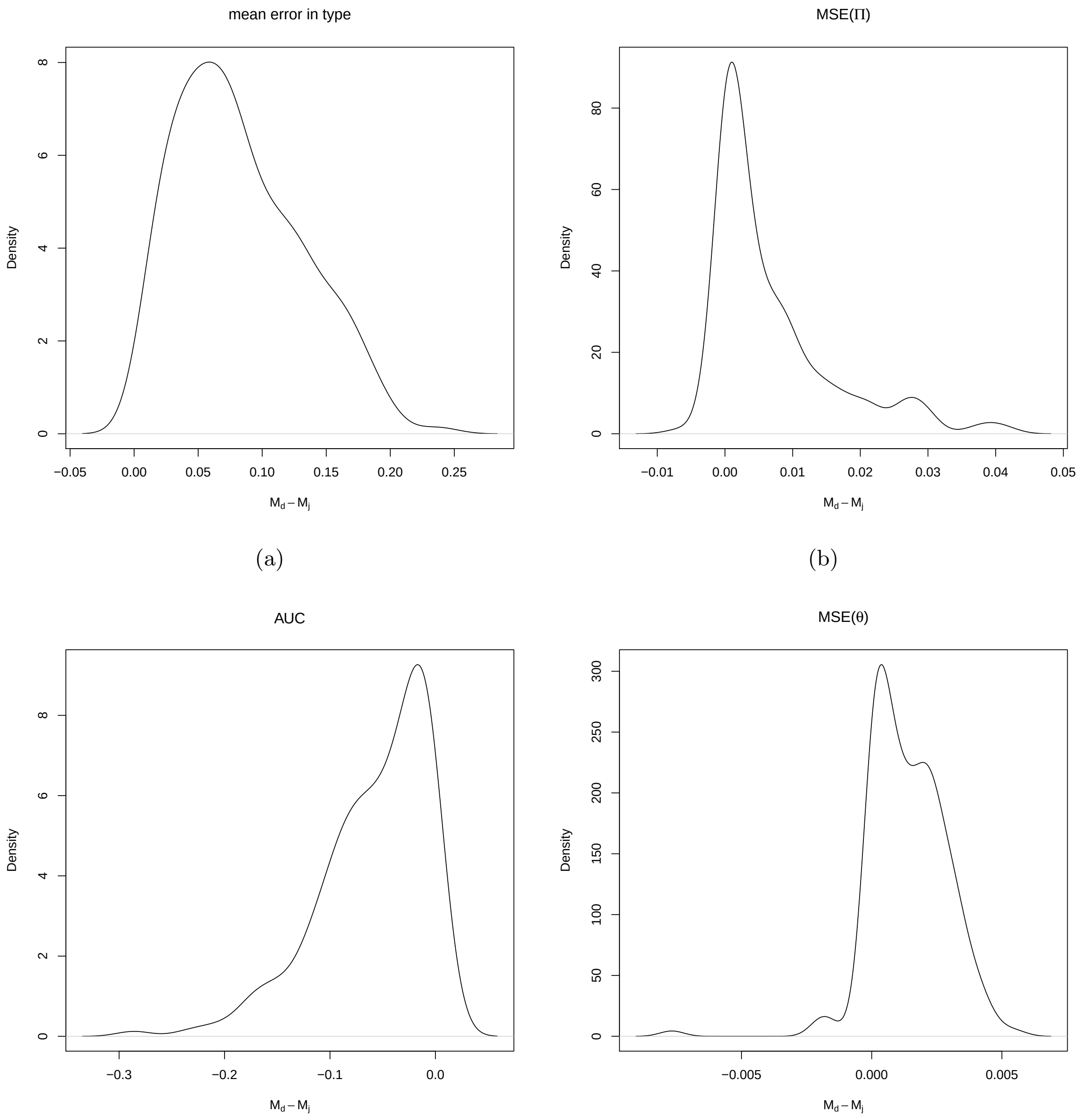
Results

Simulated Data

To test and compare the algorithm described in the previous section with the decoupled estimator, we simulated data two hundred times. For each run, we use a vector of length 3 for the marginal probability $p = \{0.3, 0.3, 0.4\}$ of each of 3 types of "article". True types for A "articles" are simulated directly from these marginal probabilities. We then construct K conditional probability matrices $\pi^{(k)}$ of size 3×3, one for each "annotator". The value of $\pi_{ij}^{(k)}$ gives the probability that annotator k annotates an article of type i with label j. We also simulate observed word terms w for each article using a matrix, θ, of conditional probabilities of words occurring in each type of article.

Two hundred such simulated data sets were analysed and for each set the biases were randomly sampled uniformly over the range 0.1 to 0.5 and split evenly between the two wrong types with the balance allocated to the correct type. This was done identically for all simulated annotators which is equivalent to having a single random annotator performing multiple annotations and the number of these annotators was sampled uniformly between 2 and 6. The words were assigned a type according to p and the word-type probabilities θ were 0.1 to appear in an article of opposite type and 0.8 to appear in an article of the same type.

Both models are then evaluated on four performance metrics:



(a)

(b)

(c)

(d)

Figure 1: Sample densities of performance measurements across multiple simulations. 200 runs of the simulated dataset analysis were performed and the mean error of type T 1(a), mean squared error of bias π 1(b), AUC 1(c) and mean squared error of word association θ 1(d) were calculated for both the joint estimator (M_j) and the decoupled estimator (M_d) for each run.

1. The mean error in expectation of type:

   $$\frac{\sum_a (1 - \mathbf{E}[T_{ai}])}{A} \tag{7}$$

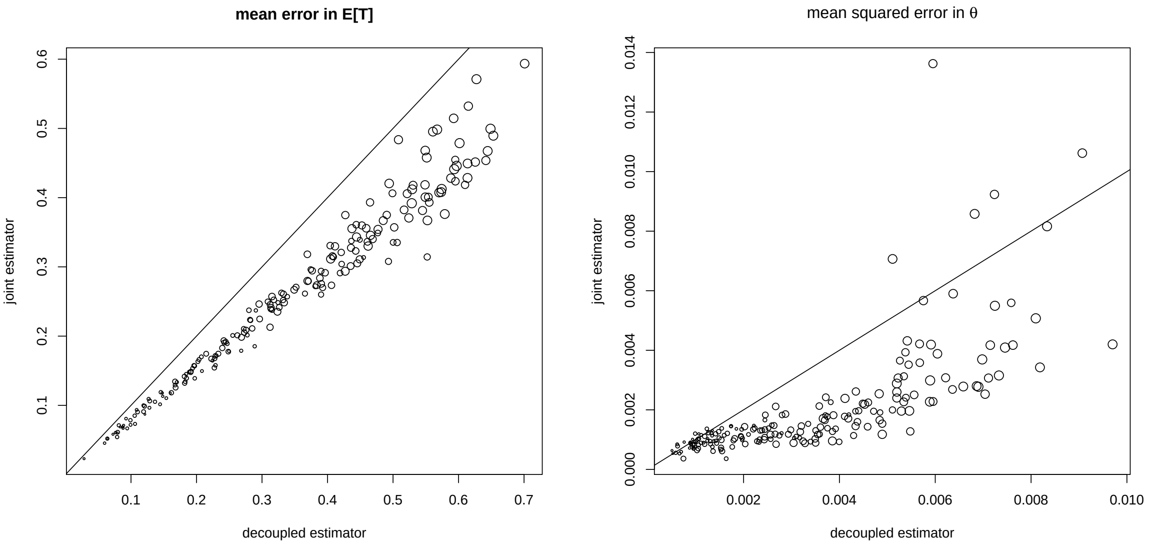
   where the true value of article a is type i.

2. The mean squared error from the π matrix of bias probabilities.

3. The mean area under the ROC curve (AUC) for each of the 3 possible types.

4. The mean squared error from the θ matrix of word-type probabilities.

We subtracted the above four statistics under the joint estimation model M_j from the decoupled estimation model M_d for repeated simulations. The mean paired difference between the above performance measures were $0.193, 0.009, -0.103$ and 0.009, respectively. All four were strongly statistically significantly different from zero under a t-test for the paired differences with p-values all less than 2.2×10^{-16}. Figure 1 shows sample density plots of these differences for the above statistics across the 200 simulation runs. Figure 2 indicates that the joint estimator's increase in performance is greater for higher biases. The size of the circles in the plot is proportional to the sampled bias and each circle represents a single run.



(a)

(b)

Figure 2: Comparison of performance across 200 iterations of simulated data. 2(a) shows the mean error in type T (as per Equation (7)) and 2(b) shows the mean squared error in word-to-type association θ. The size of the circles in the plot is proportional to the bias and each circle represents a single run. Lines with unit slope are added for reference.

Sentiment Data

We have demonstrated the superiority of the joint estimation model over the model of [3] followed by a classification step for simulated data. See [2] for an analysis of the benefit of a joint estimation model (albeit with a different classifier) in the context of medical imaging annotation. We next present our results on the sentiment dataset.

For the model fit to the full sentiment dataset, the decoupled estimator changes 13 (1.46%) article types from the majority vote model and the joint estimation model makes 90 (10.09%) changes. The results of the previous section demonstrate that the joint estimator is the preferable method. The Interquartile Range for the bias matrices was $\{0.110, 0.517\}$, indicating a level of bias comprable to the simulated dataset. Finally, Figure 3 depicts tag clouds for word terms that have the strongest negative and positive predictive power, under the joint estimation procedure. These tag clouds appear to show sensible word term associations to both positive and negative sentiment; for example, the names of the finance minister and the new agency to deal with toxic debt (NAMA) are included in the negative tag cloud.We welcome discussion regarding any interesting inclusions or omissions in these tag clouds.

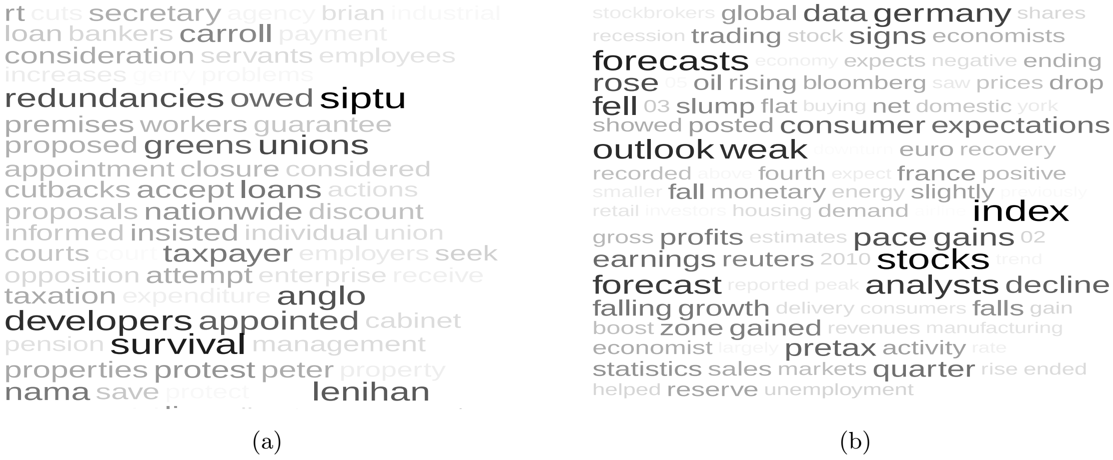


Figure 3: Tag clouds for the top 100 word terms most strongly associated with 3(a) negative and 3(b) positive articles. Most of the words appear to have an intuitively correct association with article type.

Discussion

We have demonstrated that, in the context of analysis of sentiment in media, the joint estimation model makes use of the word term association with article type and thus outperforms the decoupled model for both bias estimation and classification. This boost in performance is related to the ratio of information in the features to the biases. If the annotators are all in agreement and are unbiased then the classification will contribute little to the model. If there is bias in the annotations and the word terms are influenced by the article type then they will have a larger impact on the model and the joint estimation model will outperform the decoupled estimation model.

The joint estimator can achieve a target level of accuracy in article labelling using fewer biased annotators than the decoupled or first stage only estimator. This suggests that our method could be used to generate savings in the context of crowdsourcing with inexpert or otherwise biased annotators. Future work will include modelling a different number of categories than reported by the annotators.

REFERENCES

[1] S. Rogers, M. Girolami, and T. Polajnar. Semi-parametric analysis of multi-rater data. *Statistics and Computing*, 20:317–334, 2010. doi:10.1007/s11222-009-9125-z.

[2] V.C. Raykar, S. Yu, L.H. Zhao, G.H. Valadez, C. Florin, L. Bogoni, and L. Moy. Learning From Crowds. *Journal of Machine Learning Research*, 11:1297–1322, 2010.

[3] A.P. Dawid and A.M. 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.

[4] A. Brew, D. Greene, and P. Cunningham. Using Crowdsourcing and Active Learning to Track Sentiment in Online Media. In H. Coelho, R. Studer, and M. Wooldridge, editors, *ECAI 2010 - 19th European Conference on Artificial Intelligence*, pages 1–11. IOS Press, 2010.

[5] A. Brew, D. Greene, and P. Cunningham. The interaction between supervised learning and crowdsourcing. In *NIPS Workshop on Computational Social Science and the Wisdom of Crowds*, 2010.