Assessing the reliability of crowdsourced labels via Twitter

Noor Jamaludeen, Vishnu Unnikrishnan, Maya S. Sekeran, Majed Ali, Le Anh Trang, and Myra Spiliopoulou

University of Magdeburg, Germany
noor.jamaludeen@ovgu.de, vishnu.unnikrishnan@ovgu.de, maya.santhira@st.ovgu.de, majed.ali@ovgu.de, anh1.le@st.ovgu.de, myra@ovgu.de

Abstract. Crowdsourcing has been recently a popular solution to overcome the high cost of acquiring labeled datasets. However, the reliability of crowdsourced labels remains a challenge. Many approaches rely on domain experts who are scarce and expensive. In this work, we propose to use Twitter to acquire labels and to juxtapose them with crowdsourced ones. This allows us to measure annotator reliability. Since annotator expertise may vary, depending on content, we propose a new topic-based reliability measurement approach. We compare our model with Kappa Weighted Voting and Majority Voting as baseline methods, and show that our approach performs well and is robust when up to 30% of the annotators is not reliable.

Keywords: crowdsourcing, kappa weighted voting, annotator reliability

1 Introduction

Building a robust classification model requires a labeled dataset. Crowdsourcing for annotations has been gaining popularity over recent years. Platforms such as AmazonTurk [1] and CrowdFlower [2] offer to pay people for providing annotations. However, the quality of the annotations still needs to be checked against labels acquired from domain experts. Hence, there is a need to measure annotator reliability.

We introduce a new approach that collects labels for tweets from Twitter, organizes them on topic and assesses the reliability of the annotators with respect to the labels they assign to the tweets, taking the topics into account. In our approach, we take advantage of the fact that people are spending about two hours a day on these Social Media platforms and the amount of time spent is on a steady increase [3]. On Twitter alone, according to Statista [4], there are 335 million monthly active users.

[1] https://www.mturk.com/

[2] https://www.figure-eight.com/

[3] https://www.statista.com/statistics/433871/daily-social-media-usage-worldwide/

[4] https://www.statista.com/statistics/282087/number-of-monthly-active-twitter-users/

Our contributions are as follows. We propose a new annotation tool for tweet sentiment labeling, that capitalizes on topic-specific expertise of Twitter users. We derive topics from the tweets and use them to derive topic-based reliability scores for the annotators. These scores we use in a weighting scheme for the annotated tweets. This allows us to exploit the fact that an annotator may be more reliable for tweets belonging to a certain topic than to other topics.

This work is organized as follows. We next discuss related work on crowdsourcing and annotator reliability. In Section 3 we present the components of our approach. Section 4 contains our evaluation framework, which encompasses also a simulator for annotators. In Section 5 we report on our experiments for various percentages of unreliable annotators, as generated by our simulator. The last section concludes our study with a summary and future issues.

A note on terminology: Throughout this work, we use the terms “instance” and “tweet” interchangeably. We term a user who assigns a “label” to an instance as “annotator” and call this activity “annotation”.

2 Related Work

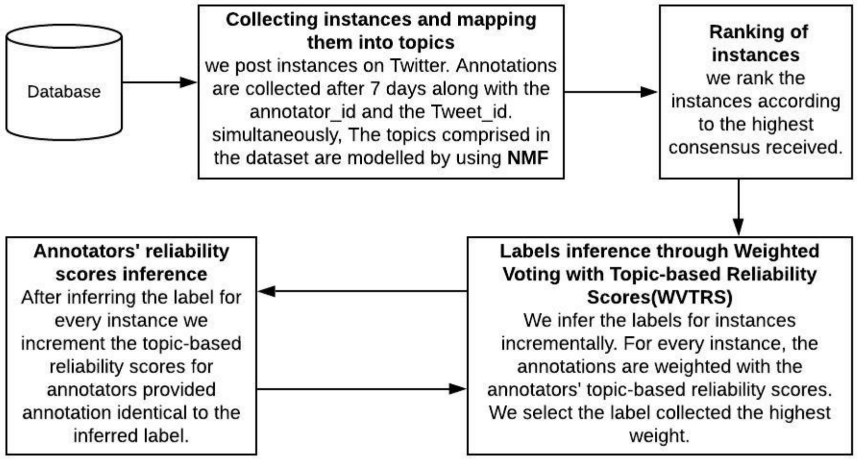
There are various approaches to tackle annotator reliability when crowdsourcing for labels [1], [2], [3]. However these studies require domain experts to validate the labels collected from annotators. In [1], Hao et al. model annotators reliability based on their cumulative performance. However, they do not consider the possibility of not having the same annotator providing labels over time. In [2], Bhowmick et al. propose a coefficient to measure annotator reliability where multiple labels can be assigned to an annotation. Here, we will be using a single label for every tweet.

Close to our work is the method of Swanson et al. [4] where annotators who have high agreements with other annotators are given higher reliability scores. In our work, annotators who deliver annotations identical to the inferred labels are assigned high reliability scores over the topics comprised in the annotated tweets. In [5], Pion-Tonachini et al. use Latent Dirichlet Allocation to model the annotators’ expertise over the classes which are analogous to topics in the topic-modeling application, in which is common to apply LDA. They define vote-class relationship to model the annotators’ individual interpretation of the classes given the votes. In our work, we do not limit the annotators’ expertise over only the classes, we learn the annotators’ reliability on latent topics modeled over the dataset, which simulates the real world setting more.

Furthermore, Pion-Tonachini et al. [5] present CL-LDA-BPE an extension model to incorporate prior knowledge of the annotators’ expertise through a structured Bayesian framework. We rather assume no prior knowledge, and therefore induce the annotators expertise from the annotations only.

3 Our Approach

Our goal is to acquire reliable sentiment labels for tweets, using Twitter users as annotators. Our approach towards this goal encompasses following tasks, depicted on Figure 1 and described in the next subsections. (1) Collecting instances and mapping them into topics, (2) Ranking instances on consensus among annotators, (3) Topic-based reliability model for the annotators, and (4) **W**eighted **V**oting with **T**opic-based **R**eliability **S**coring mechanism (**WVTRS**).



Collecting instances and mapping them into topics
we post instances on Twitter. Annotations are collected after 7 days along with the annotator_id and the Tweet_id. simultaneously, The topics comprised in the dataset are modelled by using **NMF**

Ranking of instances
we rank the instances according to the highest consensus received.

Labels inference through Weighted Voting with Topic-based Reliability Scores(WVTRS)
We infer the labels for instances incrementally. For every instance, the annotations are weighted with the annotators' topic-based reliability scores. We select the label collected the highest weight.

Annotators' reliability scores inference
After inferring the label for every instance we increment the topic-based reliability scores for annotators provided annotation identical to the inferred label.

Fig. 1: Project Workflow

3.1 Collecting instances and mapping them into topics

For the database of tweets Y (with L denoting the cardinality of Y) we acquire class labels (in our experiments: labels on sentiment) from Twitter: we developed a tool where each $y \in Y$ is posted to Twitter as a poll for a period of 7 days, during which users of Twitter can vote for one of the possible labels. The nature of the environment automatically limits users to voting only once. Once the poll has expired, every response to the tweet by each user x is stored as $(y, x, vote(y, x))$, where $vote(y, x)) \in C$ and C is the set of classes . The annotators constitute a set X, denoting its cardinality as M.

We learn the topics over Y by computing the TF-IDF values for all terms, building an instance-term matrix, and decomposing it into Instance-Topic matrix and Topic-Term matrix using Non-negative Matrix Factorization (NMF). According to the Topic-Term matrix, each term is assigned to the topic, in which the term has its maximum value. When that term occurs in a tweet, we consider this maximum value as the contribution of the corresponding topic in the tweet and we refer to it as $TP_{y,j}$. In case many terms belonging to the same topic occur in the same tweet, then the $TP_{y,j}$ is the sum of these terms-topic's maximum values. We represent each tweet y as an N-dimensional vector, as $\boldsymbol{y} =< TP_{y,1}, TP_{y,2}, ..., TP_{y,N} >$, where N is the number of topics.

3.2 Ranking instances on consensus

In most real-life crowdsourcing scenarios without monetary remunerations, it is reasonable to expect that very few users will contribute consistently to the system, giving skewed intensities with which users interact with a system. It is also possible that some instances receive more votes than others for a variety of reasons (ease of annotation, skewed availability of expertise, etc.). To accommodate this fact, we first sort the tweets on 'maximum consensus', and then step through the collected responses one tweet at a time, incrementally updating the annotator reliability (which is computed as described in the next subsection).

For tweet y and class label c, let $votes(y,c)$ be the number of annotators who assigned c to y. We assign each tweet to the class according to the majority voting, i.e. $mvlabel(y) = argmax_{c\in C} votes(y,c)$.We use this number also to assign a *rank* to y: We rank the instances in list W on how often they receive the class label $mvlabel(y)$ assigned to them. The instance with the largest number of votes takes rank position 1. This can be achieved by computing for each y the value $\frac{1}{\max_{c\in C} votes(y,c)}$ and sorting the instances accordingly. The rank reflects the agreement of annotators concerning the selected class label according to the majority voting labelling. We consider consensus as indicator of how much the class label of the instance can be trusted, and process high-ranked instances before low-ranked instances when computing annotator reliability (see next subsection).

3.3 Topic-based reliability model for annotators

To distinguish reliable annotators from unreliable ones, we introduce the concept of *reliability score*: for each tweet $y \in W$ annotated by x, we set

$$agreement(x,y) = \begin{cases} 0, & \text{if } vote(x,y) \neq inferredlabel(y) \\ 1, & \text{if } vote(x,y) = inferredlabel(y) \end{cases} \qquad (1)$$

Then, we define the reliability score of annotator x over topic j as $RS_{x,j}$, where $RS_{x,j} = \sum_{y\in W \wedge TP_{y,j}\neq 0} agreement(x,y)$. Each annotator is represented as N-dimensional vector. The j^{th} vector position contains reliability score of topic j, for $j = 1\ldots N$.

$RS_{x,j} \in [1, n_j + 1]$ where n_j is the number of tweets comprising topic j. We consider annotator a more reliable than annotator b in topic j, if $RS_{a,j} > RS_{b,j}$, i.e. annotator a provided more annotations identical to the inferred labels than annotator b did for tweets comprises topic j. A high topic-based reliability score indicates the annotator's high reliability over that topic. In the next subsection, we will refine the computing scheme of the reliability scores by taking the incremental processing of instances into account.

3.4 Weighted Voting with Topic-based Reliability Scores

Here we introduce our unsupervised incremental learning approach that applies topic-based weights to the given annotations. The votes are weighted with the annotators' topic-based reliability scores without considering the different proportions of topics comprised in a tweet. We only consider the incidence of topics.

Let W be the set of the ranked tweets. The tweets $y \in W$ are processed incrementally and the reliability scores are updated simultaneously. Here, we refine the computing scheme of the reliability scores introduced earlier in subsection 3.3. The computing will be applied in an incremental mode. We start processing with top-1 instance in list W, we infer the label of this instance using the initial reliability scores, update the reliability scores for topics comprised in the top-1 instance according to its inferred label, then move to infer the label of top-2 instance employing the updated reliability scores, reupdate again the reliability socres accordingly and so on, till we reach the last element top-N instance in the list W. The approach is described in the following steps:

1. Initialize the reliabilities for all (x, j) pairs to 1.
$$RS_{x,j,1} \leftarrow 1 \tag{2}$$
2. We infer labels for tweets incrementally, starting the inference process at the instance at rank 1. Each vote is weighted with the sum of the annotator tweet-related topic reliability scores.
$$voteWeight(x, y) \leftarrow \sum_{TP_{y,j} \neq 0} RS_{x,j,t-1} \tag{3}$$
3. The weights are aggregated for annotators who provided identical votes by summing them up.
$$classWeight(c, y) \leftarrow \sum_{vote(x,y)=c} voteWeight(x, y) \tag{4}$$
4. We select the class label that collected the highest weight as the label for the tweet:
$$InferredLabel(y) \leftarrow argmax_{c \in C}(classWeight(c, y)) \tag{5}$$
5. For each annotator who gave a vote identical to the inferred label, increment the tweet-related topic reliability scores by 1 as follows:
$$RS_{x,j,t} \leftarrow RS_{x,j,t-1} + 1 \tag{6}$$
Whereas the reliability scores for other annotators remain the same:
$$RS_{x,j,t} \leftarrow RS_{x,j,t-1} \tag{7}$$

Repeat the steps from (2) to (5) for the next tweets in the ranked list W, until all tweets in the list are processed.

The steps on how we infer the labels and derive reliability scores for annotators are detailed in the Algorithm 1.

```
INPUT:
  X: set of annotators, W: set of ranked tweets, J: set of Topics
  C: set of classes, R: set of topic-based reliability scores
  $TP_{y,j}$: contribution of topic $j$ in tweet $y$

//Initialize all topic-based reliability scores
for $x \in X$ do
    for $j \in J$ do
        $RS_{x,j} \leftarrow 1$
    end
end
for $y \in W$ do
    for $c \in C$ do
        $classWeight(c, y) \leftarrow 0$
        for $x \in X$ do
            if $label(x, y) \neq 0 \wedge vote(x, y) = c$ then
                foreach $j \in J$ do
                    if $TP_{y,j} \neq 0$ then
                        $classWeight(c, y) += RS_{x,j}$
                    ;
                end
            ;
        end
    end
    //choose the class that collected the highest weight to be the label of tweet y
    $InferredLabel(y) \leftarrow argMax_{c \in C}(classWeight(c, y))$
    // update the topic-based reliability scores
    for $x \in X$ do
        if $vote(x, y) = InferredLabel(y)$ then
            foreach $j \in J$ do
                if $TP_{y,j} \neq 0$ then
                    $RS_{x,j}$ ++
                ;
            end
        ;
    end
end
OUTPUT: Inferred labels and annotators topic-based reliability scores
```

Algorithm 1: Weighted voting with topic-based reliability scores(WVTRS)

4 Evaluation framework

To evaluate our approach we propose the metrics presented in subsection 4.1. We do not have ground truth on topic reliability. Therefore, we built a simulator, described in subsection 4.2.

4.1 Experiment Evaluation Metrics

As basis of our evaluation, we consider accuracy, computed as the ratio of correctly labeled tweets to all tweets. We further introduce an error rate metric that computes the difference between estimated and true reliability score:

$$Error_{TopicReliabilityScores} = \sqrt{\frac{\sum_{i=1}^{M}\sum_{j=1}^{N}(simRS_{x,j} - RS_{x,j})^2}{M * N}} \tag{8}$$

where $simRS_{x,j}$ are the reliability score values created by the simulator introduced in the next subsection; they serve as ground truth.

The Kappa weighted voting method [4] and the majority voting baselines do not employ topic reliability scores in the inference process. Therefore, we use the preliminary computing scheme of the reliability score introduced in subsection 3.3, in which the computation of the reliability scores is not conducted incrementally.

4.2 Simulation

Due to the difficulty of collecting labeled tweets in a closed setting for the purpose of this study, our experiment setting involves simulating annotations as we are using Social Media to collect labels. In this project, we simulate three different types of annotators; reliable, partially reliable and unreliable annotators. Reliable and partially reliable annotators represent humans with good intentions.

We refer to annotator's reliability accuracy for annotator x over topic j as $RA_{x,j}$, where $RA_{x,j} \in \{0,1\}$. Reliable annotators are more likely to deliver correct labels than partially reliable ones, hence, we assign high topic-based reliability accuracy $RA_{x,j} =$ 0.8 to reliable annotators and relatively low topic-based reliability accuracy $RA_{x,j} =$ 0.05 to partially reliable annotators. For each topic, we generate 75% of annotators to be reliable while the remaining 25% are assumed to be partially reliable. Unreliable annotators are assumed to always provide wrong labels with $RA_{x,j} = 0.0$.

To simulate the likelihood of responding to a tweet, we assume that the number of annotations each annotator provides is a random variable follows a uniform distribution. In the simulator component, we incorporate the different proportions of topics comprised in the tweets when we compute the probability of annotator x to label tweet y correctly as the weighted average with the tweet-topic coefficients of the topic-based reliability accuracy as per the formula below:

$$ProbabilityOfCorrectLabel_{y,x} = \frac{TP_{y,1} * RA_{x,1} + TP_{y,2} * RA_{x,2} + .. + TP_{y,N} * RA_{x,N}}{TP_{y,1} + TP_{y,2} + TP_{y,3} + ... + TP_{y,N}} \tag{9}$$

For every tweet y and annotator x, annotations are generated according to the likelihood of responding to tweet y and to the probability of correctly labeling it.

After the simulation of annotations, we assign to every annotator x a reliability score $simRS_{x,j}$ over topic j; they serve as ground truth. These reliability scores are computed in a similar manner to the preliminary computing scheme of the reliability scores introduced earlier in subsection 3.3. However, instead of relying on the inferred labels, the $simRS_{x,j}$ are computed based on the generated annotations and *label*(y) with respect to the known ground truth .

For each tweet y annotated by x, we set

$$simAgreement(x,y) = \begin{cases} 0, & \text{if } vote(x,y) \neq label(y) \\ 1, & \text{if } vote(x,y) = label(y) \end{cases} \tag{10}$$

Then, we compute the reliability score of annotator x over topic j as $simRS_{x,j} = \sum_{y \in Y \wedge TP_{y,j} \neq 0} simAgreement(x,y)$. Each annotator is represented as N-dimensional vector. The j^{th} vector position contains reliability score of topic j, for $j = 1 \ldots N$. $simRS_{x,j} \in [1, n_j + 1]$ where n_j is the number of tweets comprising topic j.

5 Experiments

We ran several experiments to investigate how the number of labels an annotator provides and the reliability of this annotator affects the quality of a model that classifies the instances on sentiment.

5.1 Outline

We run our experiments on the U.S. Airline Sentiment dataset [5], which we denote as A(irline) thereafter. From it, we created three random samples of size 1000, three of size 2500 and three of size 5000 tweets to be annotated. Whenever we report quality in the experiments, we refer to accuracy, averaged over the three samples of the same size.

We first run experiments to find the best number of topics to be used for our approach (subsection 5.2). Then, we tested the effect of consensus ranking on the performance of our model (subsection 5.3), assuming 500, 1000 and 2000 annotators.

To evaluate the robustness of our model we simulated three types of crowds A, B, C, in which we incorporated different percentages of unreliable annotators: 1) Crowd A: 30% of the annotators are unreliable. 2) Crowd B: 10% of the annotators are unreliable. 3) Crowd C: only reliable annotators. We used these crowds to study the effect of retaining the annotators' reliability scores in the system across many annotation tasks, assuming that a subset of annotators is active and assigns labels for several annotation tasks on the annotation platform (subsection 5.4). To test the effect of learning the annotators' reliability scores on the performance, we conducted a comparison over two aspects:(1) different number of annotators.(2) different number of annotations per annotator. (subsection 5.5).

Finally, in subsection 5.6 we report the overall performance of our model against the baselines Kappa weighted voting [4] and the Majority Voting for different number of tweets and varying number of annotators.

Across all experiments discussed earlier, the accuracy reported is the average accuracy computed over three disjoint sets of tweets.

5.2 Experiment on organizing the tweets into topics

In this experiment, we study how the number of topics affects the performance. We assume 1000 tweets and 1000 annotators, 10% of whom are assumed to be unreliable. We find that having 15 topics modeled over the entire dataset gave us the best performance as shown in Figure 2.

[5] https://www.kaggle.com/crowdflower/twitter-airline-sentiment

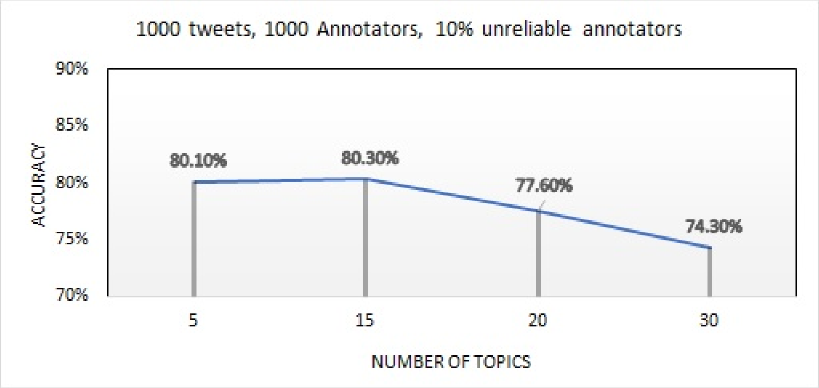


Fig. 2: Accuracy achieved as we vary the number of topics N=5,15,20,30 when considering 1000 tweets and 1000 annotators assuming 10% of them are unreliable annotators

5.3 Experiment on instances ranking

In this experiment, we study how ranking of instances improves the model performance. Due to the complete absence of any prior knowledge about the annotators, their reliability scores are estimated only from the provided annotations. Based on our assumption that the majority is reliable and since tweets are processed sequentially, we test the impact of processing the tweets that received the highest consensus first. Ranking of instances gives better estimation of the reliability scores, hence, it improves the model performance. Detailed results of comparing between ranked and unranked tweets is shown for 1000 tweets annotated by 500 annotators along different types of crowd in Table 1.

Nb of Tweets	1000		
Nb of Annotators	500		
Crowd type	A	B	C
Ranked tweets	52	65.9	69.9
Unranked tweets	50.9	65.3	69.8

Table 1: Model % accuracy by ranking of instances

5.4 Model performance for constantly active annotators

To test the performance of the model in this scenario, we simulate the time factor by assuming that annotating five datasets, where each dataset consists of 1000 tweets, is equivalent to annotating one set with 5000 tweets. Annotating two datasets of 1000 tweets per set is equivalent to annotating one set of 2000 tweets. For every dataset the annotators labeled four random tweets. As shown in Table 2, the best results are observed when annotators participated in more annotation tasks(i.e five dataset).

Nb of Annotations per annotator	4		
Nb of Annotators	500		
Nb of annotation tasks	5 tasks	2 tasks	1 task
Crowd type A	54.3	54.2	49.3
Crowd type B	68.4	67.1	64.1
Crowd type C	73.6	72	70.4

Table 2: Model % accuracy for 500 annotators labeling randomly four tweets along 5, 2, 1 tasks

5.5 Comparison of performance achieved by different number of annotators annotating randomly varying number of tweets

Assuming a fixed number of annotations is given by each annotator, the larger the number of annotators participated, the higher the accuracy achieved. A better performance was recorded for a larger set of annotators (1000 annotators) compared to the group of 500 annotators annotating randomly four tweets. Whereas a higher accuracy was recorded when the group of the 500 annotators annotated more tweets (eight tweets). However, the best performance was delivered by the smallest group of annotators (500 annotators) labeling the largest dataset with 5000 tweets according to the results shown in Table 3.

Nb of annotators	Nb of tweets	Nb of Annotations per annotator	Crowdtype A	Crowdtype B	Crowdtype C
500	1000	4	49.3	64.1	70.4
		8	58.1	75.2	79.9
	5000	24	54.3	68.4	73.6
1000	1000	4	55.6	72	79.1
	5000	12	53.9	67.4	72
2000	5000	6	50.4	66.5	71.3

Table 3: WVTRS % accuracy

5.6 Overall Performance

We compare our approach **WVTRS** against the baselines Kappa Weighted Voting **KWV** and Majority Voting **MV**. The overall results according to different number of annotators processed for Dataset A are as shown in Table 4.

Nb of annotators	Crowdtype	A			B			C		
500	Nb of tweets	1000	2500	5000	1000	2500	5000	1000	2500	5000
	Nb of Annotations per annotator	4	12	24	4	12	24	4	12	24
	WVTRS	49.2	54.2	54.3	64.1	67.1	68.4	70.4	72	73.6
	KWV	46.6	49.3	49	61.9	62.4	63.4	68.6	69.7	70.2
	MV	45.2	45.1	43.1	59.6	59.2	59.5	67.2	67.1	68.3
1000	Nb of tweets	1000	2500	5000	1000	2500	5000	1000	2500	5000
	Nb of Annotations per annotator	2	6	12	2	6	12	2	6	12
	WVTRS	50.7	50.3	53.9	61.7	66.6	67.4	70.1	72.6	72
	KWV	47.6	47.2	48.9	61.1	62.8	64.4	69.4	69.8	70.1
	MV	47.2	43.7	44.8	59	60.3	60.4	68.9	67.8	67.2
2000	Nb of tweets	1000	2500	5000	1000	2500	5000	1000	2500	5000
	Nb of Annotations per annotator	1	3	6	1	3	6	1	3	6
	WVTRS	45.8	51.8	50.4	62	64.7	66.5	69.6	72.4	71.3
	KWV	44.5	48.1	47	60.4	62.3	63.2	68.9	69.4	70
	MV	43.3	45.1	43.2	59.1	59.5	60.2	67.8	67.3	67.6

Table 4: Overall model % accuracy compared to the baselines Kappa-Weighted Voting and the Majority Voting methods

Across all the experiments, our approach performed the best compared to the baselines. The model achieved the best performance when the smallest number of annotators (500) annotated a dataset with 5000 tweets. As a result, the more annotations the annotator delivers, the model's capacity of estimating the annotator's reliability scores improves, thus the labels inference enhances. The model was also robust across different percentages of unreliable annotators and performed better than the Kappa Weighted Voting approach. We experimented our approach on a dataset which has very high homogeneity level of the comprised topics, further tests are required to determine if our model performs better with datasets that are more heterogeneous. These results suggest that the WVTRS approach that we propose gives promising results that can be augmented with testing the approach on different datasets with different levels of heterogeneity in the topics contained in the instances and more informative topics.

6 Conclusion

In this paper we propose an approach to distinguish between reliable and unreliable annotators over topical areas and to infer the labels through a weighted voting with annotators' topic-based reliability scores. We believe there is potential for our approach to improve the accuracy by improving the topic modeling step. The limitations of our approach are: 1) The different proportions of topics comprised in a tweet are treated equally. The votes are weighted with the annotators' topic-based reliability scores without considering the different proportions of topics comprised in the tweet. Due to the homogeneity of topics in the chosen dataset, the experiments do not manifest the impact of this limitation. 2) Processing the tweets online is not feasible, due to the tweets ranking step.

As future work we intend to work on the following aspect: Incorporate prior knowledge about the annotators through crawling their Twitter profiles. We can consider each annotator as a document, then apply topic modeling over tweets and annotators. Hence, we can measure the similarity between annotators and tweets and weigh the votes given by annotators with these similarities. The higher the similarity between an annotator and a tweet, the more reliable that annotators' annotation for that tweet is.

Acknowledgement

This work is supported by the German Research Foundation (DFG) under project OSCAR (Opinion Stream Classification with Ensembles and Active learners).

References

1. S. Hao, S. C. H. Hoi, C. Miao, and P. Zhao, "Active crowdsourcing for annotation," in *2015 IEEE/WIC/ACM International Conference on Web Intelligence and Intelligent Agent Technology (WI-IAT)*, vol. 2, pp. 1–8, 2015.
2. P. K. Bhowmick, P. Mitra, and A. Basu, "An agreement measure for determining inter-annotator reliability of human judgements on affective text," in *Proceedings of the Workshop on Human Judgements in Computational Linguistics*, HumanJudge '08, pp. 58–65, 2008.
3. P.-Y. Hsueh, P. Melville, and V. Sindhwani, "Data quality from crowdsourcing: A study of annotation selection criteria," *Proceedings of the NAACL HLT 2009 Workshop on Active Learning for Natural Language Processing*, pp. 27–35, 2009.
4. R. Swanson, S. Lukin, L. Eisenberg, T. Corcoran, and M. Walker, "Getting reliable annotations for sarcasm in online dialogues," in *Proceedings of the Ninth International Conference on Language Resources and Evaluation (LREC-2014)*, pp. 4250–4257, 2014.
5. L. Pion-Tonachini, S. Makeig, and K. Kreutz-Delgado, "Crowd labeling latent dirichlet allocation," *Knowledge and Information Systems*, vol. 53, pp. 1–17, 2017.
6. V. C. Raykar and S. Yu, "Ranking annotators for crowdsourced labeling tasks," *NIPS*, vol. 24, pp. 1809–1817, 2011.
7. S. Nowak and S. Rüger, "How reliable are annotations via crowdsourcing: A study about inter-annotator agreement for multi-label image annotation," in *Proceedings of the International Conference on Multimedia Information Retrieval*, MIR '10, pp. 557–566, 2010.
8. S. Rosenthal, N. Farra, and P. Nakov, "SemEval-2017 task 4: Sentiment analysis in twitter," in *Proceedings of the 11th International Workshop on Semantic Evaluation (SemEval-2017)*, pp. 502–518, 2017.
9. T. Hashimoto, T. Kuboyama, and B. Chakraborty, "Topic extraction from millions of tweets using singular value decomposition and feature selection," in *2015 Asia-Pacific Signal and Information Processing Association Annual Summit and Conference (APSIPA)*, pp. 1145–1150, 2015.