Effectiveness of Human-in-the-Loop Sentiment Polarization with Few Corrected Labels

Ruhaila Maskat[1]*, Nurzety Aqtar Ahmad Azuan[2], Siti Auni Amaram[3], Nur Hayatin[4]
Faculty of Computer and Mathematical Sciences, Universiti Teknologi MARA, Shah Alam, Selangor, Malaysia[1, 3]
Faculty of Computing, College of Computing & Applied Sciences, Universiti Malaysia Pahang[2]
Informatics Department, University of Muhammadiyah Malang, Jawa Timur, Indonesia[4]

***Abstract*—In this work, we investigated the effectiveness of adopting Human-in-the-Loop (HITL) aimed to correct automatically generated labels from existing scoring models, e.g. SentiWordNet and Vader to enhance prediction accuracy. Recently, many proposals showed a trend in utilizing these models to label data by assuming that the labels produced are near to ground truth. However, none investigated the correctness of this notion. Therefore, this paper fills this gap. Bad labels result in bad predictions, hence hypothetically, by positioning a human in the computing loop to correct inaccurate labels accuracy performance can be improved. As it is infeasible to expect a human to correct a multitude of labels, we set out to answer the questions of "What is the smallest percentage of corrected labels needed to improve prediction quality against a baseline?" and "Would randomly selecting automatic labels for correction produce better prediction than specifically choosing labels with distinct data points?". Naïve Bayes (NB) and Decision Tree (DT) were employed on AirBnB and Vaccines public datasets. We could conclude from our results that not all ML algorithms are suited to be used in a HITL environment. NB fared better than DT at producing improved accuracy with small percentages of corrected labels, as low as 1%, exceeding the baseline. When selected for human correction, labels with distinct data points assisted in enhancing the accuracy better than random selection for NB across both datasets, yet partially for DT.**

***Keywords*—Human-in-the-loop; few labels; sentiment polarization**

I. Introduction

Sentiment is used in an array of applications, from sentiment analysis to customer intelligence. Two primary sentiment polarities are "positive" and "negative". In the absence of any sentiment, a "neutral" polarity would be given. To date, additional polarities have been introduced to include the element of intensity. They are "strongly positive" and "strongly negative". Typically, the approach to determining polarity is via a hand-crafted lexicon of sentiments. Words in the lexicon were earlier identified to express positive or negative opinions about a particular subject of interest. This approach is limited to the list of collected words. Continuous enrichment must occur to sustain an ever-expanding vocabulary, especially on social media platforms with the existence of new words e.g., "Google" and "tweet". On its own, this approach may not be perpetually effective when more data are added.

Evolving from this approach is the use of machine learning (ML) algorithms trained on a dataset labelled with polarity. Through this approach, the algorithms learn from the labels and predict the polarity of newly unseen data points. This frees arduous efforts in the upkeeping of a lexicon. However, the performance of this approach relies considerably on a large number of good-quality labels which usually are produced with the assistance of human annotators. Several downsides of this technique are human annotators can be scarce in some domains, the quality of the annotators can differ depending on their level of knowledge and experience, not all datasets can be annotated, engaging human annotators are costly and they are incapable of annotating a tremendously large number of data points.

Recent works [1]–[5] show the implementation of a hybrid solution where a lexicon is used to automatically label a dataset which is then used as a training set by a ML algorithm. The assumption made is the produced labels are close to the ground truth, thus reliable. Our work investigates this assumption based on two lexicon-based scoring models of different types – a valence-based lexicon, SentiWordNet, and a rule-based lexicon, Vader. Human annotators were employed to examine the generated sentiment polarity of two public datasets (AirBnB and Vaccines) when the resulting labels conflict with one another. We found that Vader identifies polarity more like a human annotator than SentiWordNet. We extended this finding to further explore the effect Human-in-the-Loop (HITL) has on accuracy. By having a human expert in the loop, incorrect labels that compromised quality can be emended. Consequently, the ML algorithm learns the correction and updates its knowledge space, resulting in enhanced accuracy. As human annotators would not be able to correct a multitude of labels, we added another component to this experimental setting, i.e., few labels, which is counterintuitive to many ML algorithms as they often need numerous labelled data.

In this work, we determined the quality of sentiment labels from SentiWordNet and Vader based on a human's opinion. We constructed a HITL experimental framework and addressed two important questions. 1) What is the smallest percentage of corrected labels needed to improve prediction quality against a baseline? 2) Would randomly selecting automatic labels for correction produce better prediction than specifically choosing automatic labels with distinct data points?

This paper is outlined as follows. The literature review is covered in Section II. In Section III, we describe our proposed HITL methodology and formulated two research questions to be addressed via experiments. The results of the experiments are presented and discussed in Section IV. Finally, in Section V, we conclude this paper.

II. Literature Review

A. Sentiment Labelling

Traditional sentiment labelling requires human annotators to label data. Although this is known to produce gold standard datasets, unfortunately, it can be error-prone, time-consuming, labour intensive and infeasible with big data [6]. Automatic labelling comes into the picture to overcome this limitation. As of late, a scoring model such as SentiWordNet [7] or Vader [6] has been employed for this task. SentiWordNet works by utilizing a lexicon of synonymous English words clustered together a.k.a. synset. It contains 147,306 synsets labelled with the polarity of positivity, negativity and neutrality. An advantage of SentiWordNet as compared to older lexicons, e.g., Linguistic Inquiry and Word Count (LIWC), is its account of sentiment valence to assist in portraying sentiment intensity [6]. Nonetheless, SentiWordNet also shares the disadvantages of other lexicons: a shortage in coverage, where essential features tend to be missed and costly maintenance [6]. Conversely, Vader is newer and leverages rules to decide on the polarity and intensity of sentiments [6]. Vader adopted the same human-validated sentiment lexicons as LIWC, Affective Norms for English Words (ANEW) and General Inquirer (GI) yet appended more features specific to social media text, making it sensitive to their nuances. The cornerstone of Vader is the adoption of the wisdom-of-the-crowd to determine sentiment valence and the leveraging of Grounded Theory to formulate rules customized to generalize across an array of grammatical and syntactical functions for sentiment polarity decision-making.

B. Human-in-the-Loop (HITL)

HITL is not a new idea. Including feedback from a human during a computer-related process to improve effectiveness has been earlier proposed – e.g., harnessing paid feedback in crowdsourcing via Amazon Turk [8]. Another example is in the Pay-as-You-Go dataspace where users supply feedback to assist in resolving entities during data integration [9]. The HITL idea is also adopted in Few-Shot Learning [10], Active Learning [11], Transfer Learning [12] and User Guidance [13]. The underlying notion is performance can be enhanced even with a small amount of data from humans, especially for datasets that are too large to the extent of being infeasible to manually annotate [14].

C. Machine Learning Algorithms

Naïve Bayes and Decision Tree are popularly used in numerous sentiment analysis literature. To date, they include [1], [15]–[19].

1) Naïve Bayes (NB) when used for sentiment prediction is simple and intuitive yet highly accurate [20]. To predict, NB leverages the concept of conditional probability. Its advantage is it does not require a large training set [21]. However, a primary disadvantage is its tendency to theorize linguistic features to be independent under soconditionsion [20]. The argument put forth is the nature of words is co-occurring and are joined by syntactic as well as semantic dependencies, hence, NB may produce unfavourable performance.

2) Decision Tree (DT) is hierarchical in its natural form. Rules are generated for the task of predicting target terms. Each leaf node contains a word feature of the sentences in a corpus. DT was reported to have a great deal of adaptability to large datasets as compared to other ML algorithms [21]. A larger dataset triggers the formulation of additional rules, permitting the construction of higher quality trees with a larger pool of attributes available [22]. Nonetheless, this can be disadvantageous in a setting with few labels.

III. Research Methodology

A. Automatic Sentiment Production

In this step, we produced labels on each data point using automatic techniques. Two popularly used techniques are SentiWordNet and Vader scoring models. We applied both techniques to AirBnB[1] and Vaccines[2] public datasets. AirBnB contains 142,114 reviews on various premise owners in Asheville, North Carolina, United States, while the Vaccines dataset consists of 24,075 tweets related to opinions of Covid-19 vaccines worldwide. Both techniques' output is a pair of polarity confidence values for each data point; one for positive sentiment and the other for negative sentiment. The higher confidence value between the two becomes the final suggested sentiment. E.g., the positive polarity confidence for data X is 25% whereas the negative polarity confidence is 75%. Therefore, data point X will have a negative polarity.

When the set of sentiment labels is placed in tandem, three sets of results exist. The first is where both models scored positive sentiments on each data point. The second set contains only negative sentiments, and the last set has contradicting sentiments. For example, Vader scored a positive sentiment whereas SentiWordNet scored a negative sentiment on the same data point Y. The earlier two sets are not interesting in this study since both techniques produced the same result. We assume there is strong evidence to support the generated degree of confidence which led to the same result and thus it is more worthwhile to bring our focus to the contradicting set as this would allow us to discriminate between the two scoring models. For AirBnB, the size of the contrasting set was 3,967. For the Vaccines dataset, a total of 3,055 tweets were found.

B. Human Labelling

Next thing is to determine which contradicting labels are correct, Vader's or SentiWordNet's. Known as the ground truth, this task must be performed by the participation of a human. We employed two human annotators to evaluate each suggested sentiment in the contradicting set. The annotators are not English native speakers but have more than five years of writing and speaking English at a university level. We aim to investigate which model is less aligned with a human's

[1] http://insideairbnb.com/get-the-data.html
[2] https://www.kaggle.com/gpreda/all-covid19-vaccines-tweets

judgement. This model is then used to discover the role of HITL with few corrected labels in improving the performance of prediction.

The result in Table I displays Vader produced 85.73% labels that match with human labels from the AirBnB dataset. From the Vaccines dataset, 22.37% of Vader's labels match with human labels. In contrast, for the AirBnB dataset, SentiWordNet obtained 2.14% matches and 4.45% for the Vaccines dataset. In summary, SentiWordNet scored the farthest to a human's perception of sentiment on both datasets. Additionally, for the conflicting set, we discovered that SentiWordNet tends to score text as negative sentiment more than positive, whereas Vader was the opposite. This could be due to Vader's capability to handle social media features as the datasets contain online reviews of that nature. Furthermore, both the datasets are general, and thus it would be interesting to use an inherently negative dataset in the future, for example from the mental health domain, to see how both models would behave. Therefore, with these two datasets, Vader produced better scores compared to SentiWordNet.

TABLE I. SCORING MODELS MATCH WITH HUMAN LABEL

AirBnB		***Vaccines***	
Vader	**SentiWordNet**	**Vader**	**SentiWordNet**
85.73%	2.14%	**22.39%**	4.45%

C. Calculate Baseline Effectiveness

Since SentiWordNet and Vader show contradictory results to human labels, we then proceed to calculate baseline effectiveness i.e. the accuracy that could be achieved. Fig. 1 shows the processes involved. Each label set from SentiWordNet and Vader was used to train two popularly used ML algorithms, Decision Tree and Naïve Bayes. Cross-validation of five folds was employed. Afterwards, predictions of each dataset were generated. To know the accuracy of these predictions, we compare them against human labels.

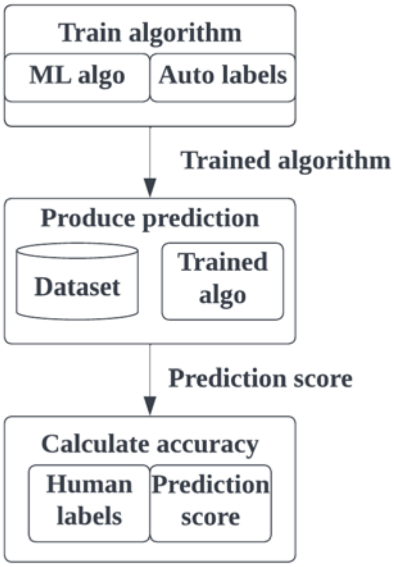

Fig. 1. Process of Calculating Baseline Effectiveness.

TABLE II. ACCURACY VALUES OF BOTH SCORING MODELS WITH SENTIWORDNET CHOSEN AS THE BASELINE

	Decision Tree (DT)		***Naïve Bayes (NB)***	
	AirBnB	**Vaccines**	**AirBnB**	**Vaccines**
SWordNet	**1.34%**	**0.00%**	**0.00%**	**0.00%**
Vader	86.54%	26.84%	87.88%	26.84%

Table II displays the baseline accuracy values. SentiWordNet labels produced substantially low accuracy across both datasets and both ML algorithms with values of 0.00% and 1.34%. Conversely, Vader labels achieved good accuracy for the AirBnB dataset of 86.54% (DT) and 87.88% (NB); the average for the Vaccines dataset with 26.84% (both DT and NB). To conclude, SentiWordNet's low-quality labels are less effective in a prediction task.

From the findings, it is interesting to learn the effects on prediction accuracy when a human expert is present to provide explicit feedback in the form of corrections to automatic labels. Hence, we performed several experiments to investigate.

D. Experimental HITL Framework Construction

To conduct this experiment, we developed a basic HITL framework and adapted it to incorporate human explicit feedback (Fig. 2). Our study did not lean towards any specific variants of HITL, alternatively, our interest is in exploring the generic idea of including humans in the labelling process to correct a small number of labels and observe the outcome. The framework consists of five layers: data, automatic label generation, algorithm training, prediction performance evaluation and label correction.

In the first layer, datasets are introduced into the framework. From these datasets, the scoring models will automatically generate labels. These scoring models are in-built with their own lexicon and thus do not require any annotated data point. Then, a human expert checks a small number of the produced labels and corrects them, if deemed necessary. By choosing to correct the labels or otherwise, the expert inadvertently injects explicit feedback into the framework. The output, a set of labels, becomes the training set for one or more machine learning algorithms. Once trained, the algorithms will produce predictions. The quality of these predictions, in the form of accuracy, is calculated to measure the effectiveness of the corrected automatic labels. A threshold of preferred accuracy is checked and if this threshold was not reached, the predicted labels are presented to the human expert for correction. Several iterations would occur until the threshold is met. Alternatively, if the human expert decided not to continue correcting anymore labels, the loop would end. This manifests the role of a human in the prediction loop; thus, the term human-in-the-loop framework.

The following questions are addressed in this work and the answers are explained in the next section.

Q1: What is the smallest percentage of corrected labels needed to improve prediction quality against a baseline?

Q2: Would randomly selecting automatic labels for correction produce better prediction than specifically choosing automatic labels with distinct data points?

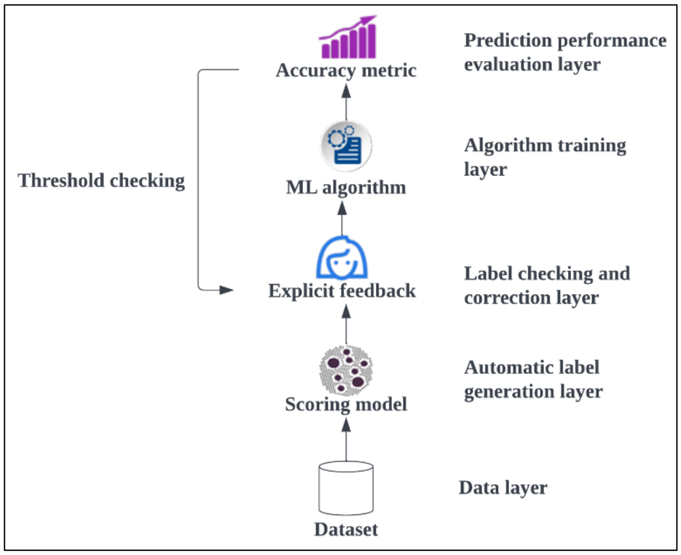

Fig. 2. Experimental HITL Framework.

IV. Result and Discussion

In this section, we describe the experimental setup and present as well as discuss the results and findings of our study. RapidMiner and Orange were employed as the simulation platform for these experiments.

A. *Experiment 1*

Aim: This experiment aims to answer Q1 where we want to determine at what percentage of corrected labels introduced in the loop would enhance accuracy. The threshold accuracy values are based on the baseline results in Table II. Since SentiWordNet yields accuracy values worse than Vader, therefore, it was used as the scoring model in this study.

Setup: We experimented with different percentages of human labels. Very small percentages of 10% and below were tested, followed by percentages of 20% and 35%. The labels were randomly selected. Two classic ML algorithms were used, Decision Tree (DT) and Naïve Bayes (NB). Cross validation of 5 folds was employed.

Result: Referring to Table III and Fig. 3, we found the following:

1) *Baseline test:* Only NB exceeded the baseline accuracy for both datasets (NB AirBnB – 99.14; NB Vaccines – 69.56). DT surpassed the baseline accuracy for the Vaccines dataset at 73.16, however, failed for AirBnB, reaching the highest accuracy of only 1.34 which is at par with the baseline.

2) *Small percentages test:* NB was able to achieve improvement in accuracy at even 1% of corrected labels for both AirBnB and Vaccines datasets. In contrast, DT did not improve for AirBnB but showed marked improvement (i.e. 73.16) only when 35% of corrected labels were supplied.

In summary, not all ML algorithms are suitable for HITL with few corrected labels. Nevertheless, with the right algorithm, a percentage of corrected labels as small as 1% can be effective in significantly enhancing accuracy. Depending on the characteristics of the dataset, further increment of accuracy beyond the initial value can occur early as witnessed from AirBnB when NB is employed. A jump in accuracy of 46.63 from 12.13 was attained at 9%. To note, these labels were chosen randomly, therefore, an interesting alternative is where a more deliberate strategy is tested. Experiment 2 explores this idea.

B. *Experiment 2*

This experiment investigates if by carefully choosing automatic labels with distinct data points would yield better accuracy than when the labels were picked randomly. The result of this experiment answers Q2. Distinct data points represent unique cases within the data space of a particular domain. This approach trains ML algorithms using a set of "small data" [23]. Small data supports the notion of quality over quantity. Here, a small number of high-quality data points that represents the majority of a population is more preferred than a large, primarily uniformed, collection of data points. Such uniformity can cause the algorithm to become blindsided and thus focuses only on a specific case, limiting its learning experience.

TABLE III. SentiWordNet Accuracy Values

	Decision Tree (DT)		***Naïve Bayes (NB)***	
	AirBnB	**Vaccines**	**AirBnB**	**Vaccines**
Baseline	1.34%	0.00%	0.00%	0.00%
1%	0.91%	0.00%	**12.12%**	**69.20%**
2%	1.01%	0.00%	12.12%	69.20%
3%	**1.34%**	0.00%	12.13%	69.20%
4%	1.08%	0.00%	12.13%	69.20%
5%	0.91%	0.00%	12.12%	69.20%
6%	1.34%	0.00%	12.13%	69.20%
7%	1.34%	0.00%	12.13%	69.20%
8%	1.34%	0.00%	12.13%	69.20%
9%	1.34%	0.00%	**46.63%**	69.20%
10%	0.66%	0.00%	81.46%	69.20%
20%	1.34%	0.00%	98.66%	69.20%
35%	1.34%	**73.16%**	**99.14%**	**69.56%**

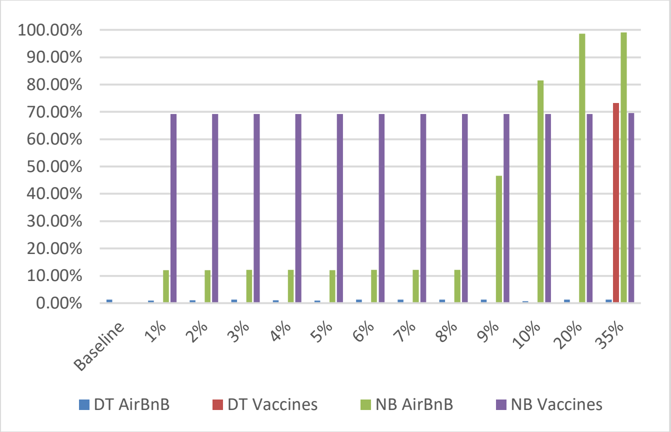

Fig. 3. SentiWordNet Accuracy Values Visualized with Bars.

To reflect the distinct nature of the data points, four techniques to calculate the distance between a pair of text were used and compared. They are Cosine, Euclidean, Jaccard and Manhattan. Afterwards, the resulting similar texts were clustered together. A dendrogram was formed and a cutting point was determined based on the production of a cluster set with a size of approximately 30 to 40 clusters. The rationale behind this condition is to produce an approximate minimum number of data points as in Experiment 1 for reasons of comparison fairness. In other words, 1% of AirBnB and Vaccines datasets, each. These data points and their labels were used to train both DT and NB. Stratification was included in this experiment to understand its possible influence on effectively producing better accuracy when there are very few human-corrected labels. Therefore, a combined total of 8 techniques were used. They are Non-Stratified Cosine (NSC), Non-Stratified Euclidean (NSE), Non-Stratified Jaccard (NSJ), Non-Stratified Manhattan (NSM), Stratified Cosine (SC), Stratified Euclidean (SE), Stratified Jaccard (SJ) and Stratified Manhattan (SM).

Setup: Alike Experiment 1, we experimented with different percentages of human labels. Very small percentages of 10% and below were tested, followed by percentages of 20% and 35%. The labels were obtained from across the derived cluster set to consist of as many unique representations as possible. Two classic ML algorithms were used: Decision Tree (DT) and Naïve Bayes (NB). Cross validation of 5 folds was employed.

Result: The following were discovered.

1) Baseline test: Generally, the results exhibit a similar pattern as found in Experiment 1, but with enhanced accuracy values in a majority of the cases. Table IV and Fig. 4 show the average accuracy of all eight techniques when no stratification was used with distinct data points. The result shows no difference in the accuracy of AirBnB when applied to DT where in both random and distinct data, DT did not surpass the baseline. In contrast, for Vaccine dataset, a higher accuracy was obtained at 78.00 while random data achieved only 73.16. With NB, it attained accuracy higher than the baseline for both datasets (NB AirBnB – 99.77; NB Vaccines – 77.79) and better than random data in Experiment 1.

When we compare to see if stratification of data can contribute to the improvement of accuracy, we found that this is true for NB but not for DT. Accuracy for AirBnB remained similarly low as the baseline i.e. 1.34 in DT. In addition, DT achieved only a slightly higher accuracy i.e. 78.07 with Vaccines dataset as compared to random data i.e. 78.00. Conversely, NB reached better accuracy in AirBnB and Vaccines datasets with stratification (NB AirBnB – 99.94; NB Vaccines – 77.85). Table V and Fig. 5 show the average accuracy for distinct data points with stratification applied.

Thus far, we can observe that in this HITL framework, taking advantage of distinct data points can produce better accuracy than just randomly selecting data for a human expert to check and correct. Additionally, applying stratification can further improve the produced accuracy. To understand the performance of each of the eight techniques, we selected the best values they produced against the baseline and the result is displayed in Table VI.

TABLE IV. AVERAGE ACCURACY FOR DISTINCT DATA POINTS WITHOUT STRATIFICATION APPLIED

	Decision Tree (DT)		*Naïve Bayes (NB)*	
	AirBnB	**Vaccines**	**AirBnB**	**Vaccines**
Baseline	1.34%	0.00%	0.00%	0.00%
1%	1.25%	0.00%	**12.12%**	**70.59%**
2%	1.26%	0.00%	12.13%	71.40%
3%	1.26%	0.00%	12.13%	71.87%
4%	1.34%	0.00%	12.13%	72.50%
5%	1.28%	0.00%	12.13%	72.96%
6%	1.34%	0.00%	12.13%	73.30%
7%	1.26%	0.00%	12.13%	73.37%
8%	1.34%	0.00%	**20.79%**	73.64%
9%	1.34%	0.00%	46.77%	73.76%
10%	1.34%	0.00%	94.32%	73.81%
20%	1.34%	**34.28%**	98.66%	75.19%
35%	1.34%	**78.00%**	**99.77%**	**77.79%**

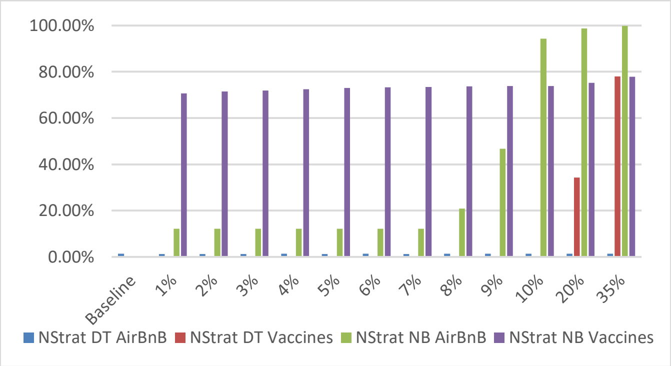

Fig. 4. Average Accuracy for Non-stratified Data Visualized with Bars.

TABLE V. AVERAGE ACCURACY FOR DISTINCT DATA POINTS WITH STRATIFICATION APPLIED

	Decision Tree (DT)		*Naïve Bayes (NB)*	
	AirBnB	**Vaccines**	**AirBnB**	**Vaccines**
Baseline	1.34	0.00	0.00	0.00
1%	1.34	0.00	**12.12**	**70.01**
2%	1.22	0.00	12.12	70.59
3%	1.28	0.00	12.13	70.82
4%	1.28	0.00	12.13	70.97
5%	1.34	0.00	12.13	71.76
6%	1.34	0.00	12.13	72.39
7%	1.34	0.00	12.13	72.48
8%	1.34	0.00	**22.45**	72.89
9%	1.34	0.00	74.33	73.09
10%	1.34	0.00	98.66	73.16
20%	1.34	**30.48**	98.66	75.36
35%	1.34	**78.07**	**99.94**	**77.85**

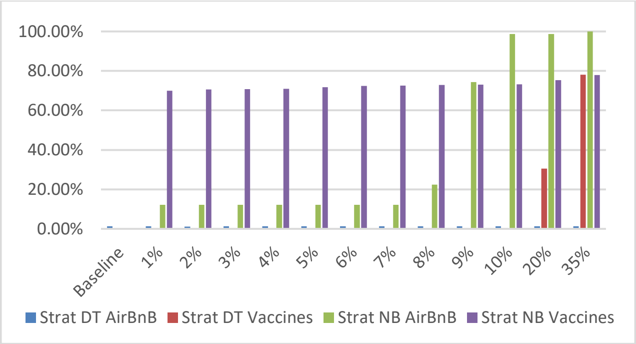

Fig. 5. Average Accuracy for Stratified Data Visualized with Bars.

TABLE VI. BASELINE TEST RESULT

	Decision Tree (DT)		***Naïve Bayes (NB)***	
	AirBnB	**Vaccines**	**AirBnB**	**Vaccines**
NSC	1.34%	78.54%	99.77%	78.33%
NSE	1.34%	77.66%	99.60%	77.44%
NSJ	1.34%	**78.88%**	99.72%	**78.68%**
NSM	1.34%	76.93%	**100.00%**	76.70%
SC	1.34%	78.01%	**100.00%**	77.80%
SE	1.34%	77.77%	**100.00%**	77.55%
SJ	1.34%	78.09%	**100.00%**	77.88%
SM	1.34%	75.90%	98.66%	75.67%

NSC-Non-Stratified Cosine; NSE-Non-Stratified Euclidean; NSJ-Non-Stratified Jaccard; NSM-Non-Stratified Manhattan; SC-Stratified Cosine; SJ-Stratified Euclidean; SJ-Stratified Jaccard; SM-Stratified Manhattan

None of the eight techniques successfully excelled over the baseline when DT was applied to the AirBnB dataset, but when NB was applied all surpassed the baseline for both datasets (Table VI). Furthermore, the top values from all the techniques were also generated when NB was used with AirBnB, even reaching 100% accuracy. For the Vaccines dataset, both DT and NB produced comparable accuracy values between 75 and 79, indicating generalization occurred well here. Jaccard, despite having stratification or otherwise, continually churned good accuracy in both AirBnB and Vaccines datasets (DT Vaccines – 78.88; NB AirBnB – 100; NB Vaccines – 78.68), but performed better with stratification.

2) Small percentages test: Similar to the result in Experiment 1, DT did not produce better accuracy for Vaccines dataset at 1% corrected labels. Nevertheless, using distinct data points resulted in the need of a smaller percentage i.e. 20% as compared to random selection i.e. 35%. Furthermore, better accuracy was obtained in this experiment i.e., 78.00 (Table IV) from 73.16 in Experiment 1 (Table III) with the same percentage of corrected label of 35%.

As with Experiment 1, NB showed better accuracy from baseline even at 1% of corrected labels for both datasets (Table IV). For AirBnB, the accuracy at 1% is identical to Experiment 1 (i.e., 12.12), yet, the improvement of accuracy occurred faster at 8% in contrast to 9% in Experiment 1. Unfortunately, this came with a cost of reduced accuracy i.e. 20.79 (Table IV). Applying stratification did not result in needing a smaller percentage different from without stratification.

In summary, supplying corrected labels with distinct data points can help in obtaining higher accuracy if coupled with a ML algorithm suitable for HITL as proven with NB on AirBnB and Vaccines datasets. Although both have different characteristics, indicating a generalized effect, more datasets will need to be tested to ascertain this. Overall, the role of stratification in affecting accuracy was positive, depending on the combination of a ML algorithm and dataset chosen. Applying an ensemble of similarity techniques can yield better result.

V. Conclusion

In this paper, we have investigated the effectiveness of using human-in-the-loop (HITL) to improve prediction accuracy by correcting automatically generated labels from existing scoring models such as SentiWordNet and Vader. As more recent work adopted the use of these scoring models in place of a training set, we took the initiative to understand if their inherent assumption of these labels being gold-standard-worthy is plausible. We experimented using two public datasets, AirBnB and Vaccines, in combination with two ML algorithms, Naïve Bayes and Decision Tree, where we discovered that Naïve Bayes produced better accuracy than Decision Tree at small percentages of corrected human labels. We also discovered that selecting labels with distinct data points to be corrected helps to enhance accuracy for Naïve Bayes but partially for Decision Tree.

Acknowledgment

The Malaysian government is funding this research through the Fundamental Research Grant Scheme (FRGS) at Universiti Teknologi MARA (UiTM) Shah Alam, Malaysia (FRGS/1/2019/SS05/UITM/02/5).

References

[1] M. Aufar, R. Andreswari, and D. Pramesti, “Sentiment Analysis on Youtube Social Media Using Decision Tree and Random Forest Algorithm: A Case Study,” 2020 International Conference on Data Science and Its Applications, ICoDSA 2020, 2020, doi: 10.1109/ICoDSA50139.2020.9213078.

[2] E. H. Almansor, F. K. Hussain, and O. K. Hussain, “Supervised ensemble sentiment-based framework to measure chatbot quality of services,” Computing, vol. 103, no. 3, pp. 491–507, 2021, doi: 10.1007/s00607-020-00863-0.

[3] J. P. Pinto and V. M. T., “Real Time Sentiment Analysis of Political Twitter Data Using Machine Learning Approach,” International Research Journal of Innovations in Engineering and Technology (IRJIET), vol. 6, no. 4, pp. 4124–4129, 2019, [Online]. Available: www.irjet.net.

[4] A. Borg and M. Boldt, “Using VADER sentiment and SVM for predicting customer response sentiment,” Expert Systems with Applications, vol. 162, p. 113746, 2020, doi: 10.1016/j.eswa.2020. 113746.

[5] M. R. A. Rahim, Y. Mahmud, and S. Abdul-Rahman, “Customers’ Opinions on Mobile Telecommunication Services in Malaysia using Sentiment Analysis,” International Journal of Advanced Computer Science and Applications, vol. 12, no. 12, pp. 222–227, 2021, doi: 10.14569/IJACSA.2021.0121229.

[6] E. Hutto, C.J. and Gilbert, “VADER: A Parsimonious Rule-based Model for,” Eighth International AAAI Conference on Weblogs and Social Media, p. 18, 2014, [Online]. Available: https://www.aaai.org/ocs/index .php/ICWSM/ICWSM14/paper/viewPaper/8109.

[7] S. Baccianella, A. Esuli, and F. Sebastiani, “SENTIWORDNET: A Publicly Available Lexical Resource for Opinion Mining,” Proceedings of the 7th Conference on Language Resources and Evaluation LREC10, pp. 417–422, 2008, [Online]. Available: http://www.lrec-conf.org/proceedings/lrec2010/pdf/769_Paper.pdf.

[8] J. Wang, S. Oyama, H. Kashima, and M. Kurihara, “Learning an accurate entity resolution model from crowdsourced labels,” Proceedings of the 8th International Conference on Ubiquitous Information Management and Communication, ICUIMC 2014, pp. 0–7, 2014, doi: 10.1145/2557977.2558060.

[9] R. Maskat, N. W. Paton, and S. M. Embury, Pay-as-you-go configuration of entity resolution, vol. 10120. 2016.

[10] K. Bailey and S. Chopra, “Few-Shot Text Classification with Pre-Trained Word Embeddings and a Human in the Loop,” pp. 1–8, 2018, [Online]. Available: http://arxiv.org/abs/1804.02063.

[11] L. Toumanidis, P. Kasnesis, C. Chatzigeorgiou, M. Feidakis, and C. Patrikakis, “ActiveCrowds: A human-in-the-loop machine learning framework,” Frontiers in Artificial Intelligence and Applications, vol. 338, pp. V–VI, 2021, doi: 10.3233/FAIA210090.

[12] L. Yang, S. Hanneke, and J. Carbonell, “A Theory of Transfer Learning with Applications to Active Learning,” Machine learning, vol. 90, no. 2, pp. 161–189, 2013.

[13] T. T. Nguyen, M. Weidlich, H. Yin, B. Zheng, Q. V. Hung Nguyen, and B. Stantic, “User guidance for efficient fact checking,” Proceedings of the VLDB Endowment, vol. 12, no. 8, pp. 850–863, 2018, doi: 10.14778 /3324301.3324303.

[14] X. Wu, L. Xiao, Y. Sun, J. Zhang, T. Ma, and L. He, “A Survey of Human-in-the-loop for Machine Learning,” 2021, [Online]. Available: http://arxiv.org/abs/2108.00941.

[15] B. Lakshmi DeviV., V. Bai, and K. Somula Ramasubbareddy Govinda, “Sentiment Analysis on Movie Reviews,” Emerging Research in Data Engineering Systems and Computer Communications, pp. 321–328, 2020.

[16] R. Ahuja, A. Chug, S. Kohli, S. Gupta, and P. Ahuja, “The impact of features extraction on the sentiment analysis,” Procedia Computer Science, vol. 152, pp. 341–348, 2019, doi: 10.1016/j.procs.2019.05.008.

[17] M. Guia, R. R. Silva, and J. Bernardino, “Comparison of Naive Bayes, support vector machine, decision trees and random forest on sentiment analysis,” IC3K 2019 - Proceedings of the 11th International Joint Conference on Knowledge Discovery, Knowledge Engineering and Knowledge Management, vol. 1, no. Ic3k, pp. 525–531, 2019, doi: 10.5220/0008364105250531.

[18] A. Bayhaqy, S. Sfenrianto, K. Nainggolan, and E. R. Kaburuan, “Sentiment Analysis about E-Commerce from Tweets Using Decision Tree, K-Nearest Neighbor, and Naïve Bayes,” 2018 International Conference on Orange Technologies, ICOT 2018, no. October, 2018, doi: 10.1109/ICOT.2018.8705796.

[19] V. A. Fitri, R. Andreswari, and M. A. Hasibuan, “Sentiment analysis of social media Twitter with case of Anti-LGBT campaign in Indonesia using Naïve Bayes, decision tree, and random forest algorithm,” Procedia Computer Science, vol. 161, pp. 765–772, 2019, doi: 10.1016/j.procs.2019.11.181.

[20] P. Gamallo and M. Garcia, “Citius: A Naive-Bayes Strategy for Sentiment Analysis on English Tweets,” 8th International Workshop on Semantic Evaluation, SemEval 2014 - co-located with the 25th International Conference on Computational Linguistics, COLING 2014, Proceedings, no. SemEval, pp. 171–175, 2014, doi: 10.3115/v1/s14-2026.

[21] J. Singh, G. Singh, and R. Singh, “Optimization of sentiment analysis using machine learning classifiers,” Human-centric Computing and Information Sciences, vol. 7, no. 1, 2017, doi: 10.1186/s13673-017-0116-3.

[22] M. Bilal, H. Israr, M. Shahid, and A. Khan, “Sentiment classification of Roman-Urdu opinions using Naïve Bayesian, Decision Tree and KNN classification techniques,” Journal of King Saud University - Computer and Information Sciences, vol. 28, no. 3, pp. 330–344, 2016, doi: 10.1016/j.jksuci.2015.11.003.

[23] J. J. Faraway and N. H. Augustin, “When small data beats big data,” Statistics and Probability Letters, vol. 136, pp. 142–145, 2018, doi: 10.1016/j.spl.2018.02.031.