On Label Quality in Class Imbalance Setting
A Case Study

Jumanah Alshehri, Marija Stanojevic, Eduard Dragut, and Zoran Obradovic

Temple University

IEEE International Conference on Machine Learning and Applications

Special Session: Machine Learning for Natural Language Processing

December 12-14, 2022

Introduction

- Producing high-quality labeled data is a challenge where in many cases, human involvement is necessary to ensure the label quality
- Human annotations are not flawless, especially in the case of a challenging problem
- Label quality is not the only challenge in in supervised learning; sampling is also a major challenge (e.g., class imbalance)
- In this work, we report several strategies to enhance the predictions in the Article-Comment Alignment Problem (ACAP)*
- In our setting, we encounter two main challenges:
  - Noisy label, caused by the high disagreement among annotators since
  - Sampling user comments to be labeled by human annotators which gives highly imbalanced datasets.

* Alshehri, J., Stanojevic, M., Dragut, E., Obradovic, Z., Stay on Topic, Please: Aligning User Comments to the Content of a News Article. ECIR 2021.

Article-Comments Alignment Problem (ACAP)*

- Finding the relevancy level between article and comments.
- Labels :
  - Relevant
  - Same Category
  - Same Entities
  - Irrelevant
- Five datasets (WSJ, TG, DM, MW, and FN), different length, number of articles and comments.
- Three annotators, labeled 1K examples per dataset

* Alshehri, J., Stanojevic, M., Dragut, E., Obradovic, Z., Stay on Topic, Please: Aligning User Comments to the Content of a News Article. ECIR 2021.

Motivation

- Article and comment pair from Daily Mail to be labeled
- BK Score (in c): represent the annotators confidence level regarding the article topic. The most accurate label is the label obtained by the third annotator (Ann 3).

Agreement Analysis

- FK = Fleiss Kappa and α = Krippendorff's alpha score
- Labeling article-comment pairs in WSJ are the most challenging task, with the smallest correspondence between the annotators
- We divided data points to Gold (GL) and Noise (NL) Labels according to inter-annotator variation σ:
  - GL: σ = 0, all annotators agree on one label
  - NL: σ > 0, at least one annotator disagrees with the other annotators
- The number of GL and NL varies across the datasets

Dataset	**WSJ**	**TG**	**DM**	**MW**	**FN**
FK	0.22	0.36	0.37	0.40	0.45
α	0.42	0.60	0.61	0.64	0.66

Dataset	**# GL**	**% GL**	**# NL**	**% NL**
WSJ	443	44%	557	56%
TG	795	80%	205	20%
DM	833	83%	167	17%
FN	858	86%	142	14%
MW	862	86%	138	14%

Proposed Strategies

1. **Pre-Weight Labeling (Pre-WL):**
   - Annotators' knowledge differs based on their interests.
   - This technique uses the annotators' background knowledge (confidence score) to obtain the final label
     - We asked annotators to scale their knowledge regarding each news topic from [1-10]
     - We weight each label given by an annotator with he/she confidence level based on he/she knowledge scale (λ).

$$\omega = \begin{cases} 0.3 & \text{if } \lambda = [1-4] \\ 0.6 & \text{if } \lambda = [5-7] \\ 1 & \text{if } \lambda = [8-10] \end{cases}$$

     - The final aggregated lal

$$\hat{y} = \sum_{i=1}^{n} (l_i \, \omega_{ic})/n$$

- $\hat{y}$ is the aggregated label
- n is the number of annotators
- l_i is the label given by the i^{th} annotator
- ω_{ic} is the confidence level for the i^{th} annotator for a given topics c

Proposed Strategies

2. Post-Weight Labeling (Post-WL)

- Utilize the inner-disagreement between annotators by allowing the model to treat each example differently during the training process according to the disagreement leve
  - We calculate σ, the inner-disagreement which is the variation per example
  - Then walculate the corresponding weight ω for each example by leveraging the exponential growth and decay concept:
  - The t

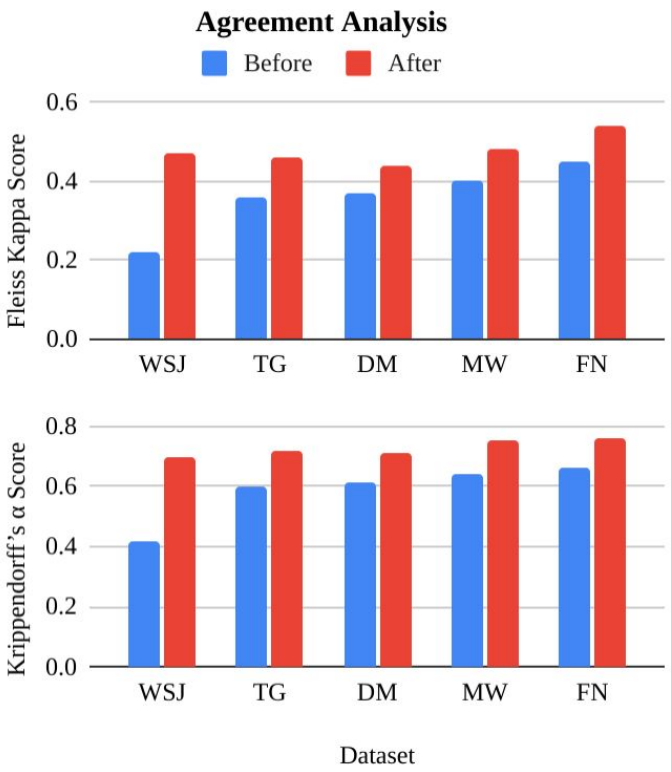
$$\omega = \begin{cases} 1 & \text{if } \sigma = 0 \\ \delta^{\sigma} & \text{if } otherwise \end{cases}$$

δ is a hyper-parameter representing the change rate. We tried different values of δ and found via experiments that 0.5 give the best performance.

Proposed Strategies

3. Annotator Relabeling

- This approach focuses on reverting to the annotators to relabel the NL examples.
  - Identifying NL examples
  - Understand annotators' common mistakes during labeling.
  - We meet with the annotators, explain the labeling mistakes
  - Ask annotators' to relabel the examples without looking into the previous noisy label.
- The inner-agreement score increased in both metrics in all datasets, especially in WSJ, where the agreement score was the lowest before relabeling.



Experiments

- **Data:**
  - Five news outlets - Wall Street Journal (WSJ), Fox News (FN), Daily Mail (DM), The Guardian (TG), and Market Watch (MW)
  - 1K labeled article-comment pairs.
  - Each class is associated with score **{4, 3, 2, 1}**, corresponding to **{Irrelevant, Same Entity, Same Category, and Relevant}** respectively.
- **Classification Model:**
  - Utilize **BERTAC***: that leverage BERT base architecture
  - We introduce the ordinal classification loss to **BERTAC**

$$weight = 1 + \frac{|\bar{y_i} - y_i|}{k-1}$$

- $k = 4$ (number of classes)
- y_i is the actual label
- $\bar{y_i}$ is the predicted label of the example

* Alshehri, J., Stanojevic, M., Dragut, E., Obradovic, Z., Stay on Topic, Please: Aligning User Comments to the Content of a News Article. ECIR 2021.

Experiments

- **Baselines:**
  - **Original:** it contains GS and NE data points, where **GS** examples represent **54%-87%** of the data, while **NE** represents **13%-46%**
  - **GS:** Only labeled data points with **perfect agreement scores**
  - **Random Labeling (RL):** In this naive strategy, we randomly assign a label for the NL examples that are different from the given noisy label. For example, if the noisy label is 1, we randomly assign 2, 3, or 4 to that example

Results: Overall Performance

	Dataset	WSJ	FN	DM	TG	MW
Baselines	Original	**86.8(.4)**	91.5(.6)	88.6(.9)	90.5(.9)	91.3(.8)
	GL	85.0(.2)	**92.9(.3)**	**89.5(.5)**	**91.2(.8)**	**92.0(.9)**
	RL	64.7(.9)	71.8(.9)	66.4(.9)	63.8(.6)	72.5(.5)
Strategies	Pre-WL	84.3(.2)	81.7(.7)	82.7(.4)	66.7(.4)	86.2(.6)
	Post-WL	80.7(.4)	83.3(.6)	74.4(.9)	73.9(.4)	84.8(.7)
	Relabel	83.1(.9)	88.0(.2)	85.0(.3)	88.7(.6)	88.4(.1)

- **RL** is a “pure luck” strategy with a probability of 33% that a particular random label matches the relevancy level between article-comment pairs
- **GL** outperforms Original, which means that the noisy labels in the Original confuse the model
- **WSJ** performance did not improve when using GL; this is because NL examples in WSJ represent more than 50% of WSJ population compared to the rest of the dataset (NL = 14%-20%)

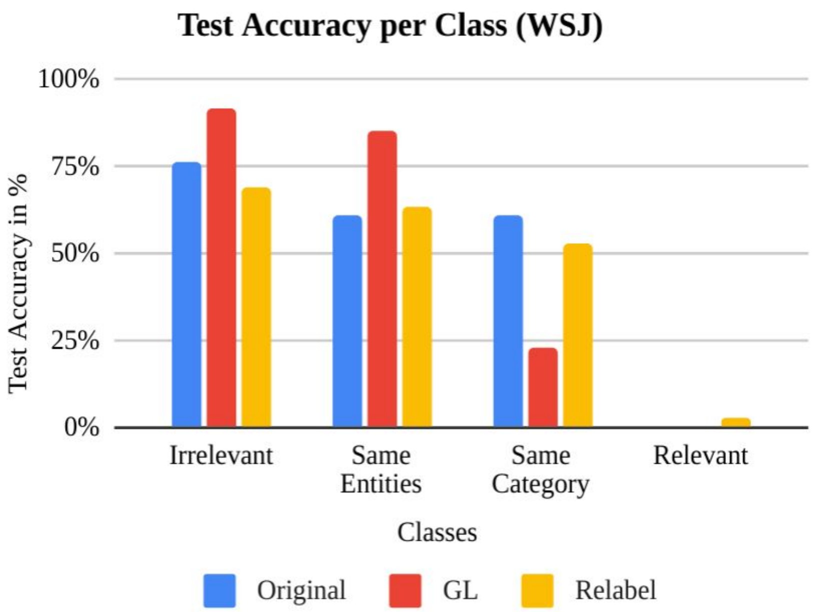
Results: Overall Performance

Dataset		WSJ	FN	DM	TG	MW
Baselines	Original	**86.8(.4)**	91.5(.6)	88.6(.9)	90.5(.9)	91.3(.8)
	GL	85.0(.2)	**92.9(.3)**	**89.5(.5)**	**91.2(.8)**	**92.0(.9)**
	RL	64.7(.9)	71.8(.9)	66.4(.9)	63.8(.6)	72.5(.5)
Strategies	Pre-WL	84.3(.2)	81.7(.7)	82.7(.4)	66.7(.4)	86.2(.6)
	Post-WL	80.7(.4)	83.3(.6)	74.4(.9)	73.9(.4)	84.8(.7)
	Relabel	83.1(.9)	88.0(.2)	85.0(.3)	88.7(.6)	88.4(.1)

- None of the proposed strategies, including relabeling, outperform Original and GL
- Although the agreement score between annotators shown increases between 9%-25% when relabeling the NL examples, the model performance in Relabel strategy declines between 2%-3% compared to the Original
- Is it reasonable to waste resources and relabel more examples?

Results: Prediction for Each Label

- Although the overall test accuracy for Original and GL is preferable, with test accuracy of 86% and 85% respectively, we can see that both strategies fail in predicting the “Relevant” class
- “Relevant” class distribution increased by 5% which allows the model to make correct predictions for that class.
- Given the distinct semantics of each class, the “Same Entities” class performance is not affected much by the class distribution. The entity name in the article-comment pair helps the model learn this class better, even in the presence of few examples.



Results: Solving Imbalance Issue

Dataset	Original	GL	Relabel	W-Loss
WSJ	86.8	85.0	83.1	**88.6**
FN	91.5	92.9	88.0	**93.0**
DM	88.6	89.5	85.0	**89.7**
TG	90.5	91.2	88.7	**91.9**
MW	91.3	92.0	88.4	**92.5**

- The previous observations change the problem directions; going back to the annotators and ask them to relabel the data is a misuse of resources in our case
- This experiment show one of the traditional data imbalance methods, Weighted Loss [*].
- Reducing the class imbalance problem with the Weighted Loss (W-Loss) method, while keeping noise labels, enhances the model performance.

* Cao, K., Wei, C., Gaidon, A., Arechiga, N. & Ma, T. Learning Imbalanced Datasets with Label-Distribution-Aware Margin Loss. NeurIPS. (2019)

Conclusion

- We analyze several strategies for enhancing human annotators' label quality for the Article-Comment Alignment Problem (ACAP)
- Our results show that despite reducing the disagreement between annotators, in the case of imbalanced data, this does not help enhance the model's performance
- We advocate that one needs to consider reducing class imbalance, in addition to allocating resources to relabeling, as this also can help enhance a model's overall performance
- In the future, we will focus on combining data imbalance methods with our label quality strategies to further enhance the predictions of ACAP
- We also plan to identify more problems with high class imbalance and noisy labels, and work through the lessons learned in this case study