AN EMPIRICAL STUDY OF MODEL GENERALIZATION UNDER DISTRIBUTION SHIFT

Selena Sun, Tony Sun, Joseph Zhang, Anthony Zhan

ABSTRACT

Machine learning models deployed in a real-world setting are vulnerable to distribution shifts. For example, models to give movie recommendations may become less accurate due to changes in audience preferences, or models to predict Amazon product ratings from reviews may be inaccurate given changing rating distributions over time. In this project, we aim to explore model generalizability under a distribution shift for the Multilingual Amazon Reviews Corpus (MARC) over time. Our results showed roughly equal performance between training on an entire history of data and a recent period, indicating that continuous retraining is likely a remedy for distribution shifts. Our dataset is publicly accessible via Google Drive, and our code is open-sourced on Github.

1 INTRODUCTION

Consider this: you want to buy a t-shirt on Amazon, and you found one that looks amazing. In fact, all the people who buy it initially rave about the product and leave great reviews. However, six months later, customers realize that the color fades, the fabric is cheaply made, and the shirt shrinks more than expected.

Following notation in Lipton et al., say that we train a model f in January to predict the rating of a review given the review title and body. Given that the t-shirt sells well initially, assume that 70% of the reviews in January are 5-star reviews (distribution P). Under this assumption, we can expect a model trained in January to perform similarly in production on a dataset with mostly positive reviews.

However, we find that by June, customers have largely left scathing reviews and are deeply unsatisfied with their purchase. Perhaps half of all the reviews are now 1-star (distribution Q). It is difficult for us to expect that our model f can peform similarly well in production given that our input data can no longer be considered i.i.d.

2 DISTRIBUTION SHIFT

Two common explanations for this distribution shift are named covariate shift and label shift. Covariate shift is related to predicting effects ($p(y|x)$ stays the same), whereas label shift is related to predicting causes ($p(x|y)$ stays the same) Lipton et al. (2018); Schoelkopf et al. (2012).

In this work, we choose to focus on label shift in the context of sentiment analysis for Amazon reviews. Label shift can be mathematically represented as the following:

$$q(y, x) = q(y)p(x|y)$$

We observe that a label shift makes sense in our scenario: (1) $p(y)$ changes: distribution of stars change over time (become more negative) and (2) $p(x|y)$ stays the same: the probability of the way a review is worded stays mostly the same given a rating (e.g. a 5-star review will still tend to contain highly positive words).

Conversely, we recognize that a covariate shift might not be applicable for our context. For example, $p(y|x)$ might change if users start to use different words to describe a product. For example, if people start to use the phrase "it's bussin"' to say that a product is great, this would be an example of a change in $p(y|x)$ since this slang was not used previously. For our dataset, we will make the assumption that people will tend still use the same words to describe a product over time.

3 DATASET

To observe model generalization under label shift, we artifically engineer a label shift in which reviews turn more negative over time in the Multilingual Amazon Reviews Corpus.

3.1 MULTILINGUAL AMAZON REVIEWS CORPUS

We use the Multilingual Amazon Reviews Corpus, a collection of 210,000 Amazon reviews (200,000 train, 5,000 dev, 5,000 test) collected between 2015 and 2019 in multiple languages: English, German, Spanish, French, Japanese, and Chinese (Keung et al. (2020)). Every item in the dataset contains the following information (a sample is shown below):

```
{
"stars": 1,
"review_title": "The product is junk.",
"review_body": "I received my first order of this
    product and it was broke so I ordered it again...",
...
}
```

3.2 ARTIFICIAL LABEL SHIFT

Since the Multilingual Amazon Reviews Corpus is static (no concept of time), we artificially engineer a label shift over a time period of six months (Figure 1). The purpose of manually creating this label shift is two-fold: (1) well-formatted production data containing a distinct label shift is difficult to obtain and (2) creating the label shift ourselves enables more fine-grained control over what we want the label shift to look like.

Specifically, we merge the train, dev, and test sets of the English subset Multilingual Amazon Reviews Corpus and extract only the review title, review body, and stars. We then concatenate the review title and review body and make it lowercase.

We create a new column called "month" that determines which month a given review belongs to. For each month, we use 3,000 reviews sampled without replacement from a pre-defined monthly distribution. For example, month 1 contains 450 $(0.15 \cdot 3,000)$ 4 and 2100 $(0.7 \cdot 3,000)$ 5-star reviews, whereas month 6 contains mostly 1 and 2-star reviews.

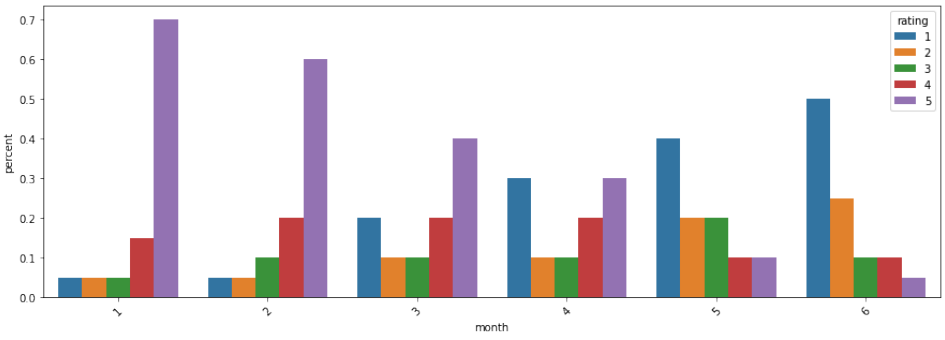


Figure 1: Label-shifted data distribution over months

4 METHODOLOGY

4.1 FASTTEXT

We use `fastText` for text classification Joulin et al. (2017). Notably, `fastText` is simple and fast, being built on top of a linear model with a rank constraint. This is similar to the continuous bag

of words model. The goal of the model is minimize the average log loss for a set of documents:

$$L = y_n \log(f(BAx_n))$$

where x_n is the normalized bag of words of the nth document, y_n is the label of the nth document, and A and B are the weight matrices being learnt Joulin et al. (2017).

The authors use some tricks such as using a bag of n-gram model over the traditional bag of words sentence representation. Additionally, the authors use hierarchical softmax to speed up the softmax operation over a large number of classes (which is especially true when the vocabulary size is large). Hierarchical softmax is based on the Huffman encoding tree, where the probability associated with a node is represented by the probability of the path from the root to the node Goodman (2001):

$$P(n_{l+1}) = \prod_{i=1}^{l} P(n_i)$$

where $P(n_i)$ is the probability of each node along the path to the root and the target node is at depth l.

4.2 Training Set Experimentation

We experimented with two different training techniques to combat the distribution shift.

Baseline: Our baseline naive training is to train on month 1 to predict all other months' ratings (e.g., we would train on month 1 data to predict month 2,3,etc.).

Cumulative Training: We train on all previous months' data, then predict the next month (e.g., train on months 1,2 to predict month 3, train on months 1,2,3 to predict month 4, etc.). This is born from the hypothesis that the distribution shift can be corrected by training on more recent data, and that generalization is still possible.

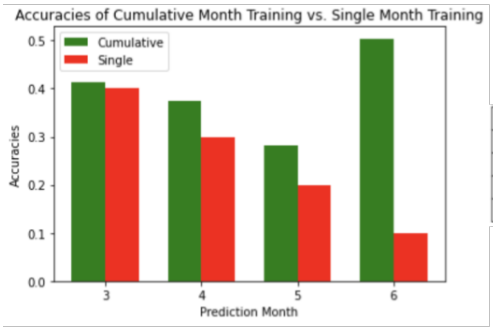
Sliding Window: We train on the previous two months' data, then predict the next month (e.g., train on months 1,2, to predict month 3, train on months 2,3 to predict month 4, etc.). Since cumulative training can be computationally expensive (for example, training on 20 years of data to predict the next month), we propose this training method to examine how generalizable training on a sliding window is.

5 Results & Discussion

All the accuracies reported below are optimized with hyperparameter tuning. We trained the FastText model over a range of learning rates, epochs, and window sizes.

5.1 Cumulative Training

With cumulative training, we achieved improved results for months 3-6 (months 2 predictions were excluded because accuracies between cumulative training and baseline are the same). The results are visualized below.

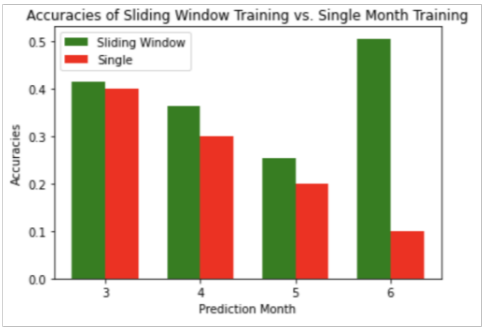


Month	Cumulative	Baseline
3	0.413	0.400
4	0.373	0.300
5	0.282	0.200
6	0.504	0.100

Figure 2: Cumulative Training Comparison Bar Chart (left) and Table (right)

5.2 2-Month Sliding Window Training

Surprisingly, we achieved almost identical results with our 2-month sliding window training as we did with our cumulative training. We suspect that we'll receive even better results from training a deeper network, or training on more data. Nevertheless, with the shallow model and our dataset size, we showed some improvement over our baseline model, especially for month 6. Results are visualized below.



Month	Sliding Window	Baseline
3	0.413	0.400
4	0.363	0.300
5	0.253	0.200
6	0.505	0.100

Figure 3: Sliding Window Training Comparison Bar Chart (left) and Table (right)

6 Conclusion

In this work, we build and open-source a new dataset based off of the Multilingual Amazon Reviews Corpus that captures label shift over time.

We find that training for a longer window tends to improve performance on the baseline result (training on just the original month). Unknown distribution shifts in the real world may occur, which necessitates retraining the model on recent data. Our preliminary results show similar accuracies for training over a sliding window and training over an entire cumulative history. This suggests that using the sliding window may be less computationally expensive and thus more feasible in a real-world setting.

In the future, we hope to experiment with a few more (and perhaps deeper) models (Word2Vec, BERT, Doc2Vec), train on a larger dataset, and try to extract principles of generalizability across different datasets.

References

Joshua Goodman. Classes for fast maximum entropy training. *CoRR*, cs.CL/0108006, 2001. URL `https://arxiv.org/abs/cs/0108006`.

Armand Joulin, Edouard Grave, Piotr Bojanowski, and Tomas Mikolov. Bag of tricks for efficient text classification. In *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume 2, Short Papers*, pp. 427–431. Association for Computational Linguistics, April 2017.

Phillip Keung, Yichao Lu, György Szarvas, and Noah A. Smith. The multilingual amazon reviews corpus, 2020. URL `https://arxiv.org/abs/2010.02573`.

Zachary Lipton, Yu-Xiang Wang, and Alexander Smola. Detecting and correcting for label shift with black box predictors. In Jennifer Dy and Andreas Krause (eds.), *Proceedings of the 35th International Conference on Machine Learning*, volume 80 of *Proceedings of Machine Learning Research*, pp. 3122–3130. PMLR, 10–15 Jul 2018. URL `https://proceedings.mlr.press/v80/lipton18a.html`.

Bernhard Schoelkopf, Dominik Janzing, Jonas Peters, Eleni Sgouritsa, Kun Zhang, and Joris Mooij. On causal and anticausal learning, 2012.