Improving Sentiment Analysis with Data Augmentation

Hang Jiang, Laura Sun
{hjian42, lasun} @ stanford.edu

Introduction

Sentiment analysis is a common Natural Language Processing (NLP) task that can be applied in many areas such as healthcare, customer services and marketing. However, the data annotation process is usually lengthy and expensive. In this project, we introduce a novel data augmentation technique to sentiment analysis. The input to our algorithm is IMDb movie reviews. We then augment the training data and apply Logistic Regression and Support Vector Machine (SVM) to output positive or negative sentiment predictions. Our study shows a consistent improvement in model performance on sentiment analysis by using data augmentation.

Data

Original Dataset

- Internet Movie Database (IMDb)[4]: 25,000 training data and 25,000 for testing (positive or negative)
- Further split the original training set into 20,000 for training and 5,000 for validation

Easy Data Augmentation (EDA) Technique[10]

- Synonym Replacement, Random Insertion, Random Swap, Random Deletion

Data Augmentation

- We conduct data augmentation only on the 20K training set such that we obtain 40K, 60K, 80K, 100K, 120K, 140K, 160K, 180K, 200K training sets

Features

Linguistic Features

- n-gram features[7]: unigram, bigram, trigrams, etc
- sentiment lexicons: Bing Liu's Opinion Lexicon[6], MPQA Subjectivity Lexicon[2], SentiWordNet[1]

World Embeddings

- GloVe[5] word vectors: 50d pre-trained word vectors with 6B tokens and 400K vocab
- To obtain each review's embeddings, we take the average of word embeddings

Data	Unigram	Bigram	Trigram	Glove	OPN	SWN	MPQA	Total
20K	70,663	1,343,826	3,004,389	50	2	1	2	4,418,933
180K	81,323	4,728,986	13,336,622	50	2	1	2	18,146,986

Table 1: This table shows the number of features for the original (20K) and augmented training data (180K) respectively with Naug=8. OPN refers to Opinion Lexicon. SWN refers to SentiWordNet.

Models

Logistics Regression

- We first run Logistic Regression model to obtain a baseline on prediction accuracy. We aim to minimize the cost function in regularized logistic regression given as:

$$J(\theta) = -\frac{1}{m}\sum_{i=1}^{m}[y_i log(h_\theta(x_i)) + (1-y_i)log(1-h_\theta(x_i))] + \frac{\lambda}{2m}\sum_{j=1}^{n}\theta_j^2$$

where $h_\theta(x_i) = \frac{1}{1+e^{-\theta^T x_i}}$We tune the inverse of regularization strength C = 0.01, 0.05, 0.25, 0.5, 1.

SVM

- A Support Vector Machine (SVM) is a large margin classifier characterized by a hyperplane separating two classes. The optimization objective is given as

$$\min\ [C\sum_{i=1}^{m}[y_i \text{cost}_1(\theta^T x_i) + (1-y_i)\text{cost}_0(1-\theta^T x_i)] + \frac{1}{2}\sum_{j=1}^{n}\theta_j^2]$$

where m is example size, n is the number of features and C is the penalty parameter. We tune the penalty parameter C = 0.001, 0.005, 0.01, 0.05, 0.1.

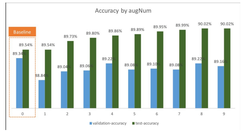
Results

Using the default Logistics Regression (LR) to train augmented data, we achieved 90.02% (+0.48%) accuracy on test set. We found the optimal augmentation number at augNum=8 (training size=180K). After fine-tuning LR and SVM on the augmented data, we achieved 90.12% (+0.53%) on LR and 90.21% (+0.44%) on SVM.

model	training size	training-acc	validation-acc	test-acc
tuned LR+GloVe	20,000	76.04%	76.42%	75.02%
tuned LR+ngrams	20,000	100%	89.38%	89.57%
tuned LR+GloVe+ngrams	20,000	100%	89.66%	**89.59%**
tuned LR+GloVe+ngrams+lexicon	20,000	100%	89.08%	89.12%
tuned SVM+GloVe	20,000	76.18%	76.28%	75.12%
tuned SVM+ngrams	20,000	99.99%	89.46%	89.64%
tuned SVM+GloVe+ngrams	20,000	100%	89.66%	**89.77%**
tuned SVM+GloVe+ngrams+lexicon	20,000	100%	88.88%	89.17%
default LR +GloVe+ngrams	20,000	100%	89.34%	89.54%
	40,000	100%	88.84%	89.54%
	60,000	100%	89.04%	89.73%
	80,000	100%	89.06%	89.80%
	100,000	100%	89.22%	89.86%
	120,000	100%	89.08%	89.89%
	140,000	100%	89.10%	89.95%
	160,000	100%	89.08%	89.99%
	180,000	100%	89.22%	**90.02%**
	200,000	100%	89.16%	90.02%
fined-tuned LR+GloVe+ngrams	180,000	100%	89.64%	**90.12%**
fined-tuned SVM+GloVe+ngrams	180,000	100%	89.46%	**90.21%**

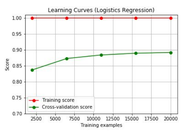
Discussion

- **How Much Augmentation**: Low augNum does not introduce enough diversity and high augNum generates too much noise to the training data. Naug =8 gives the optimal performance. This is aligned with our expectation.



1) Accuracy improvement on baseline

2) Validation loss on augmented data



3) Learning curves for the baseline

4) Accuracy by review lengths

- **Ablative Analysis**:
  - n-grams features are more important than others
  - The addition of lexicon features causes the model to overfit on training
- **Error Analysis**:
  - *Overfitting*: The baseline model suffers from high variance and low bias. The addition of augmented data helps address overfitting
  - *Out of vocabulary (OOV) words*: Many misclassified instances by the baseline model contain new words which were not present in the original data. Data augmentation allows the model to generalize to OOV
  - *Complicated structure*: One review example says "*i must admit when i first began watching this film i had no clue what was going on so the beginning was a bit confusing however that did not diminish my enjoyment...*". The current model misclassified this example due to the many negative words and changes in the polarity in the end (i.e., "however").
  - *Various sentence lengths*: The current model does not perform well on reviews with less than 70 words. Significant amount of sparse and discrete features are generated using n-grams on short reviews

Future

In the future, we can experiment with more advanced models such as Attention-LSTM[8], BERT[3], and ELMo[9] on the augmented data to see whether we can see a consistent improvement with these models. Furthermore, we can combine EDA technique and back-translation to augment the data and see whether we can achieve more improvement.

References

1. Stefano Baccianella, Andrea Esuli, and Fabrizio Sebastiani. Sentiwordnet 3.0: an enhanced lexical resource for sentiment analysis and opinion mining. In Lrec, volume 10, pages 2200–2204, 2010.
2. Lingjia Deng and Janyce Wiebe. Mpqa 3.0: An entity/event-level sentiment corpus. In Proceedings of the 2015 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, pages 1323–1328, 2015.
3. Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. arXiv preprint arXiv:1810.04805, 2018
4. Andrew L. Maas, Raymond E. Daly, Peter T. Pham, Dan Huang, Andrew Y. Ng, and Christopher Potts. Learning word vectors for sentiment analysis. In Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies, pages 142–150, Portland, Oregon, USA, June 2011. Association for Computational Linguistics.
5. Jeffrey Pennington, Richard Socher, and Christopher Manning. Glove: Global vectors forword representation. In Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP), pages 1532–1543, 2014
6. Bing Liu, Minqing Hu, and Junsheng Cheng. Opinion observer: analyzing and comparing opinions on the web. In Proceedings of the 14th international conference on World Wide Web, pages 342–351. ACM, 2005.
7. Bo Pang, Lillian Lee, and Shivakumar Vaithyanathan. Thumbs up?: sentiment classification using machine learning techniques. In Proceedings of the ACL-02 conference on Empirical methods in natural language processing-Volume 10, pages 79–86. Association for Computational Linguistics, 2002
8. Matthew E Peters, Mark Neumann, Mohit Iyyer, Matt Gardner, Christopher Clark, Kenton Lee, and Luke Zettlemoyer. Deep contextualized word representations. arXiv preprint arXiv:1802.05365, 2018
9. Yequan Wang, Minlie Huang, Li Zhao, et al. Attention-based lstm for aspect-level sentiment classification. In Proceedings of the 2016 conference on empirical methods in natural language processing, pages 606–615, 2016
10. Jason W Wei and Kai Zou. Eda: Easy data augmentation techniques for boosting performance on text classification tasks. arXiv preprint arXiv:1901.11196, 2019