Supplementary Material for W-NUT 2019 Paper: Weakly Supervised Attention Networks for Fine-Grained Opinion Mining and Public Health

Giannis Karamanolakis, Daniel Hsu, Luis Gravano
Columbia University, New York, NY 10027, USA
{gkaraman, djhsu, gravano}@cs.columbia.edu

A Sentiment Classification

In this section, for reproducibility, we discuss all details of the datasets (Section A.1) as well as the configuration of the techniques and the evaluation methodology (Section A.2) for the sentiment classification experiments.

A.1 Datasets

The Yelp'13 corpus (Tang et al., 2015) contains 335,018 user reviews of local businesses. Each review includes a 5-star rating ranging from 1 (negative) to 5 stars (positive). The IMDB corpus (Diao et al., 2014) contains 348,415 movie reviews with ratings ranging from 1 (negative) to 10 stars (positive). For both corpora, training (80%), validation (10%), and test (10%) sets are provided.

For evaluation, we use the SPOT-Yelp and SPOT-IMDB datasets. These datasets contain 100 Yelp reviews and 97 IMDB reviews from the Yelp'13 and IMDB test sets, respectively. Each dataset has been segmented both at sentences (SPOT-*-SENT) and EDUs (SPOT-*-EDU). The test sets have 3 labels (Table 1): "negative," "neutral," and "positive." For more statistics, see Tables 1 and 2 in reference (Angelidis and Lapata, 2018), as well as Table 1 in this paper.

A.2 Implementation Details

Model Parameters For a fair comparison, all the MIL-* models have the same parameter configuration as MILNET (Section 5.3 in Angelidis and Lapata (2018)). For all models using word embeddings (i.e., Seg-*, Rev-*, MIL-*), we initialize the word embeddings using 300-dimensional ($k = 300$) pre-trained word2vec embeddings (Mikolov et al., 2013). For the CNNs we use kernels of size 3, 4, and 5 words, 100 feature maps per kernel, stride of size 1, and max-over-time pooling to get fixed-size segment encodings (resulting in $\ell = 300$). For the forward and backward GRUs we use hidden vectors with 50 dimensions ($n = 2 \cdot 50 = 100$), while for the attention mechanism we use vectors of 100 dimensions ($m = 100$). We use dropout (with rate 0.5) on the word embeddings and the internal GRU states. We use L2 regularization for the softmax classifier.

Training and Validation Procedure We segment the training and validation reviews into sentences[1] and use the available review labels for training our model, over 5 classes for Yelp'13 and 10 classes for IMDB. We group the training reviews in mini-batches of 200 reviews so that reviews under the same mini-batch have a similar number of segments M. Thus, we allow for training the models using different values of M per batch while at the same time we minimize the amount of zero-padding, leading to more efficient training. As an objective function, we use the negative log-likelihood of the model parameters. We train our models using the Adadelta optimizer (Zeiler, 2012) (with learning rate 0.005) for up to 50 epochs and we stop the training process if the validation loss does not decrease for more than 10 epochs. We fine-tune the model parameters on the validation set.

Evaluation Procedure While the training and validation sets have 5 labels, for Yelp'13, and 10 labels, for IMDB, the test sets have 3 labels. During evaluation, we address this discrepancy by following the same procedure as in Angelidis and Lapata (2018) to map the segment probability distributions from 5 classes–for Yelp'13–and 10 classes—for IMDB–to 3 classes, namely, "neg-

[1] We do not segment the reviews into EDUs, because this procedure requires the use of a Rhetorical Structure Theory parser, which does not exist for every language. Instead, we opt for a language independent model. At test time, the same model is applied on both sentences and EDUs.

ative," "neutral," and "positive":

1. We map the predicted probability distribution p_i for each segment s_i into a polarity score $g_{s_i} = \sum_c p_i^c \cdot w^c \in [-1, 1]$, where $w = \langle w^1, \ldots, w^C | w^c \in [-1, 1] \rangle$. The weights w_c are spaced uniformly such that $w^{c+1} - w^c = \frac{2}{C-1}$. In particular, for the 5-class setting (Yelp) we get: $w = \langle -1, -0.5, 0, 0.5, 1 \rangle$, while for the 10-class setting (IMDB) we get: $w = \langle -1, -0.778, -0.556, -0.333, -0.111, 0.111, 0.333, 0.556, 0.778, 1 \rangle$.

2. We compute a gated polarity score $g'_{s_i} = \alpha_i \cdot g_{s_i}$, where α_i is the attention weight assigned to s_i by the model.

3. We map each score g'_{s_i} to one of the 3 discrete labels using two thresholds t_1, $t_2 \in [-1, 1]$: segment s_i is classified as "negative" if $g'_{s_i} < t_1$, "positive" if $g'_{s_i} > t_2$, and "neutral" otherwise.

We evaluate the models using the macro-averaged F1 score. We determine the value of the t_1 and t_2 thresholds using 10-fold cross-validation and report the mean scores across the 10 folds.

B Discovering Foodborne Illness

In this section, for reproducibility, we discuss all details of the datasets (Section B.1) as well as the configuration of the techniques and the evaluation methodology (Section B.3) for the experiments regarding the foodborne application.

B.1 Datasets

We use the same training and test sets as in (Effland et al., 2018). The review-level training set ("Silver" set in (Effland et al., 2018)) contains 21,551 (5,895 "Sick," 15,656 "Not Sick") reviews posted before January 1, 2017. The review-level test set contains 2,975 (949 "Sick," 2,026 "Not Sick") reviews posted after January 1, 2017. Sample weights are also calculated to account for the selection bias in this dataset (Effland et al., 2018).

To test the ability of the models to detect sentences of the "Sick" reviews discussing food poisoning, epidemiologists annotated each sentence for 437 out of the 949 "Sick" test reviews. Given a review for labeling, epidemiologists read the whole review text and decided on the label for each sentence. This led to 3,114 labeled sentences (630 "Sick," 2,484 "Not Sick"). For this application, EDU-level labels were not available, so we consider only sentences as review segments.

B.2 Implementation Details

Model Parameters For the *-BoW classifiers, the review text is encoded as a bag-of-words vector including n-grams (for n=1, 2, and 3) and each term is weighted using the Term Frequency-Inverse Document Frequency (TF-IDF) statistic (Leskovec et al., 2014). For the Rev-* and MIL-* classifiers, we use the same model parameter configuration as in Section A.2. We initialize the word embeddings using 300-dimensional pretrained word2vec embeddings.

Training and Validation Procedure We split the review-level training set into training (90%) and validation (10%) sets, randomly stratified by label and sample weight. We do not use any sentence-level labels for training. We group the training reviews in mini-batches of 200 reviews so that reviews under the same mini-batch have a similar number of segments. We train our models using the Adadelta optimizer for up to 50 epochs and we stop the training process if the validation loss does not decrease for more than 10 epochs. We fine-tune the model parameters on the validation set with respect to the F1 score.

Evaluation Procedure Given a test review, we predict a label for each sentence and aggregate the sentence predictions to get a single review prediction. For review-level classification, we use the review prediction, while for sentence-level evaluation we use the individual sentence predictions. The segment-level confidence scores are computed by multiplying the segment probability for the "Sick" class with its attention weight. To account for the selection bias in the review-level test set, we compute precision and recall using sample weights (Effland et al., 2018). Because of the class imbalance at both the review and sentence levels, we report precision, recall, F1 score, and area under the precision-recall curve (AUPR). Also, we follow Effland et al. (2018) and estimate 95% confidence intervals (95% CI) for the F1 and AUPR metrics using the percentile bootstrap method (Efron and Tibshirani, 1994) with sampled test sets of 1,000 reviews. For sentence-level classification, we also report the accuracy score.

	Review-Level Evaluation				Sentence-Level Evaluation				
Model	**Prec**	**Rec**	**F1 (95% CI)**	**AUPR (95% CI)**	**Acc**	**Prec**	**Rec**	**F1**	**AUPR**
KWRD1	0.801	0.581	0.673 (0.646, 0.699)	0.194 (0.179, 0.208)	0.850	0.806	0.342	0.481	0.408
KWRD2	0.532	0.898	0.668 (0.647, 0.689)	0.033 (0.027, 0.040)	0.890	0.778	0.640	0.703	0.572
Rev-LR-BoW	0.853	0.882	0.867 (0.852, 0.882)	0.914 (0.900, 0.929)	0.891	**0.821**	0.588	0.685	0.809
Rev-LR-EMB	0.704	0.574	0.633 (0.513, 0.714)	0.696 (0.649, 0.755)	0.797	0.500	0.843	0.628	0.489
Rev-CNN	0.803	0.898	0.848 (0.832, 0.866)	0.935 (0.923, 0.946)	0.887	0.793	0.594	0.679	0.247
Rev-RNN	0.856	0.878	0.867 (0.849, 0.884)	0.929 (0.915, 0.942)	0.913	0.810	0.745	0.776	0.113
MIL-avg	0.674	0.537	0.598 (0.485, 0.682)	0.643 (0.596, 0.708)	0.903	0.750	0.780	0.765	0.736
MIL-softmax	0.829	0.928	0.876 (0.859, 0.890)	**0.941** (0.926, 0.994)	0.912	0.755	0.833	0.792	0.816
MIL-sigmoid	**0.865**	**0.929**	**0.896** (0.882, 0.910)	0.913 (0.887, 0.926)	**0.920**	0.764	**0.874**	**0.815**	**0.840**

Table 4: Review-level (left) and sentence-level (right) evaluation results for discovering foodborne illness in Yelp reviews.

B.3 More Results and Examples

Detailed Evaluation Results Table 4 includes the evaluation results, which were reported in Table 3, as well as more baselines and evaluation metrics. For completeness, we also evaluate the "KWRD*" class of keyword search classifiers: "KWRD1" predicts the "Sick" class if the "food poisoning" phrase is included in the (lemmatized and lower cased) review text. "KWRD2" predicts the "Sick" class if at least one of the following terms are included in the review text: "food poisoning," "sick," "vomit," "diarrhea."

References

Stefanos Angelidis and Mirella Lapata. 2018. Multiple instance learning networks for fine-grained sentiment analysis. *Transactions of the Association for Computational Linguistics*, 6:17–31.

Qiming Diao, Minghui Qiu, Chao-Yuan Wu, Alexander J Smola, Jing Jiang, and Chong Wang. 2014. Jointly modeling aspects, ratings and sentiments for movie recommendation (JMARS). In *Proceedings of the 20th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 193–202.

Thomas Effland, Anna Lawson, Sharon Balter, Katelynn Devinney, Vasudha Reddy, HaeNa Waechter, Luis Gravano, and Daniel Hsu. 2018. Discovering foodborne illness in online restaurant reviews. *Journal of the American Medical Informatics Association*.

Bradley Efron and Robert J Tibshirani. 1994. *An introduction to the bootstrap*. CRC Press.

Jure Leskovec, Anand Rajaraman, and Jeffrey David Ullman. 2014. *Mining of massive datasets*. Cambridge University Press.

Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg S Corrado, and Jeff Dean. 2013. Distributed representations of words and phrases and their compositionality. In *Advances in Neural Information Processing Systems*, pages 3111–3119.

Duyu Tang, Bing Qin, and Ting Liu. 2015. Document modeling with gated recurrent neural network for sentiment classification. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 1422–1432.

Matthew D Zeiler. 2012. Adadelta: an adaptive learning rate method. *arXiv preprint arXiv:1212.5701*.