Project Report
Deep Learning Library for DPCNN - Text Categorization

Team: EntropyKnights

Abstract

Text categorization is an important task whose applications include spam detection, sentiment classification, and topic classification. In recent years, neural networks that make use of word order have been shown to be effective for text categorization. Shallow word-level CNNs have been able to outperform state-of-the-art text classification deep networks. Deepened word level CNNs can be used to capture global representations of data. The network depth of these neural architectures can be increased to get the best accuracy without increasing the computational cost by a lot. A paper [5] has been published on this idea; however, the implementation is not yet publicly available as a library. We intend to implement this paper. The datasets we will use are AG and Sogou, Dbpedia, Yahoo, Yelp, and Amazon.

1 Introduction

Motivated by success of shallow word-level CNNs over deep and complex neural networks for the important task of text categorization, Deep Pyramid Convolutional Neural Network (DPCNN) was proposed as a deeper word-level CNN in order to capture global representations of text. DPCNN outperformed the state-of-the-art methods on 6 benchmark datasets, while still remaining one of the top methods for text classification. Since no official implementation was released, and the only unofficial implementation that we could find had errors, it would be worthwhile creating a publicly available implementation of DPCNN for everyone to be able to use. We have implemented the DPCNN model all by ourselves.

2 Prior Related Work

Currently the top models for text classification are variations of BERT [4] (Bidirectional Encoder Representations from Transformers). As seen on the State of the Art (SOTA) leader boards for Amazon Review Full [1] and Polarity [2], BERT-Large is ranked one with accuracy scores of 65.83% and 97.37% respectively, while DCPNN is ranked two with scores of 65.19% and 96.68% respectively. In other leader boards there are more variations in between the two models, but the accuracy difference is usually between 1 and 2%. While BERT-Large has clear performance lead, DPCNN uses approximately 30 million trainable parameters, while BERT-Large uses approximately 330 million . Notably BERT-Base uses approximately 110 million trainable parameters. Therefore, our models strength is relatively shorter training times.

3 Approach and Model

We are implementing the DPCNN model [**DPCNN**] using the Pytorch library [3] and the torch.nn.Module subclass. The DPCNN model is illustrated in Figure 1.

3.1 Region Embedding

Since the data consisted of text entries having variable lengths, we use a custom batch generation function to generate batches and offsets for the data. This function is passed to the parameter collate_fn of torch.utils.data.DataLoader.

The network starts with region embedding in the first layer, which involves embedding of text regions covering one or more words. For the region embedding, we first obtain the text instances from Torchtext datasets. Then, we pass it to an embedding bag layer which computes the mean value of the ”bag” of embeddings. These embeddings are passed to a convolution layer with a kernel size equal to 3 to get the region embeddings.

3.2 Pre-Activation

The pre-activation section of the model consists of 2 convolutional layers with a kernel size of 3, and a

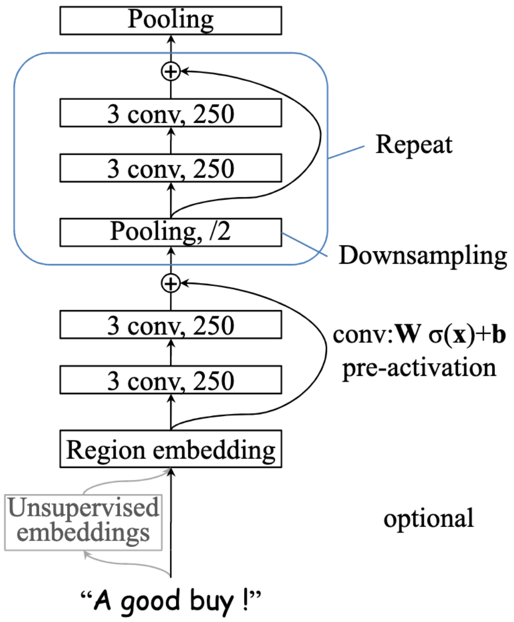

Figure 1: DPCNN Model [5]

constant 250 channels. The input to pre-activation section is added to the output of the convolutional layers to prevent vanishing gradients while training. This is referred to as a shortcut connection. Padding is used to keep the dimensions through the convolutional layers constant; and therefore, the shortcut connection simply needs to copy its single initial channel 250 times to be the correct dimension. As suggested by the name – pre-activation – there are RELU layers before both convolutional layers.

3.3 DPCNN Repeating Block

This is a repeating block consisting of a max-pooling layer followed by two convolutional layers. Max-pooling is performed with a kernel size of 3 and a stride of 2 to down sample. The convolutional layers both take in 250 channels and output 250 channels, while performing a convolution with kernel size 3, and padding 1, to maintain dimensions. The output from the max-pooling is added to the output from the second convolutional layer. We use six such blocks sequentially in our implementation. A deeper network would likely result in better results but could considerably increase memory usage and training time.

3.4 Output

The output layers are another repeated convolutional layer followed by a max-pooling layer. These two layers have the same parameters and architecture used in the repeating blocks. A fully connected linear layer follows this to transform the output to the dimensional categories that are then passed to softmax and loss layers.

4 Experiments

4.1 Datasets

This report examines a collection of well established classification datasets available through PyTorch. These include: DBpedia, Sogou News, AG News, Yelp Polarity, Yelp Full, Yahoo Answers, Amazon Reviews Full, Amazon Reviews Polarity. The validation sets of each dataset used consisted of 10k documents randomly sampled from the training set. To store the dataset between training run experiments we stored the training and testing datasets to a pickle file to allow reloading of the dataset without downloading from the Pytorch server repeatedly. Each dataset consists of a title, article description, and categorization class as a target.

The datasets vary in size with the smallest being AG News with 110K training documents, 4 classes, and an average word length of 45. The most complex dataset is Amazon Reviews Full with 3 million training documents, 5 classes, and an average word length of 93.

4.2 Implementation

To keep our scripts organized, we use a TrainBox class, which takes in training parameters and has separate functions for setting the model, setting the optimizer, training, inference, and more. Tensorboard is used to visualize recorded logs during the training epochs where accuracy and loss are collected then plotted.

4.3 Evaluation Metrics

We are using accuracy as our evaluation metric and the cross-entropy loss is minimized to optimize the model. In Pytorch the cross-entropy loss is the combination of both log softmax and negative log-likelihood loss. The cross-entropy loss was evaluated on the training samples. At each epoch the loss is evaluated against the validation set. As a final metric, we collect training, validation and testing accuracy using the all datasets. In figure

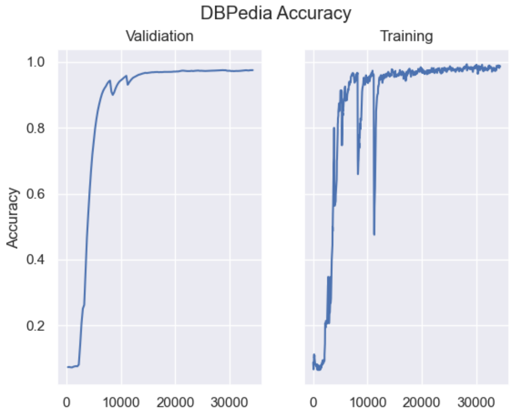

Figure 2: DBPedia Accuracy

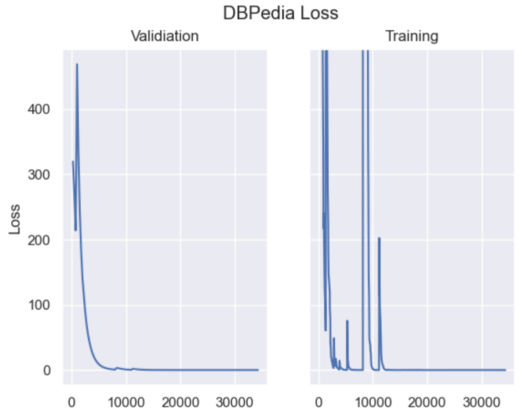

Figure 3: DBPedia Loss

2 both the training and validation set accuracy is demonstrated over training steps for the DBpedia dataset.

4.4 Hyperparameters

The DPCNN model uses several hyperparameters. We used mini-batch RMSprop [6] with a starting learning rate of 0.01 that would then decay to a rate of 0.001 afterwards (after 8th epoch if training for 10 epochs). The number of channels used throughout the network for the convolutional nets was 128 with embedding dimensions of 128. We found that a batch size of 100 was optimal for our training to take the full advantage of the GPU and also get accurate estimates for the error gradient. Lastly, depending on the dataset, the model was trained for 5-10 epochs.

4.5 Model Comparison

4.6 Results

The Loss and Accuracy graphs seen in Figures 6, 7, 2, 3, 8, 9, 4, 5, 12, 13, 10, 11 have an exponential weighted moving average applied to them using an alpha value of 0.25. The y-axis of the loss graphs is limited to the 1.05% of the max validation value, so the crucial information is not drowned out by the spikes in training loss.

The training results for the DBPedia dataset are illustrated in the figures 2 and 3.

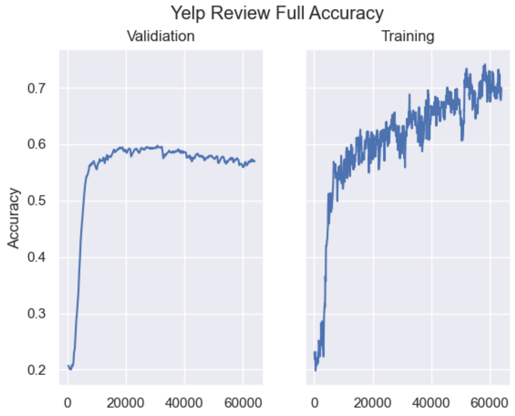

Figure 4: Yelp Review Full Accuracy

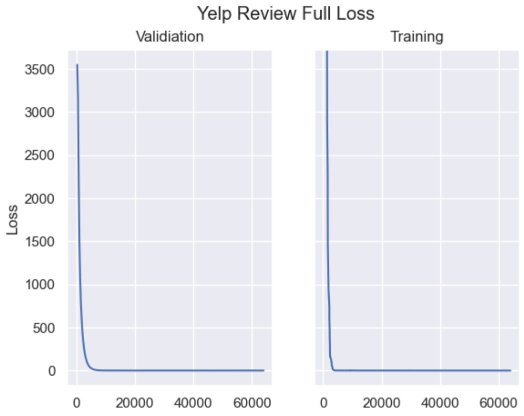

Figure 5: Yelp Review Full Loss

The total training time for the DBpedia dataset with 10 epochs was about 30 minutes. This was on a machine with RTX 2080 GPU for training on the DBpedia dataset containing 550K training documents. The learning rate and overall hyperparameters chosen for this experiment seem acceptable as for the majority of datasets the accuracy for the validation data converges over time at a high value and does not dip, which shows no signs of overfitting. The datasets in the majority have loss and accuracy graphs similar to Figures 3 and 2, and can be seen in section A. The two exceptions, Yelp Review Full and Amazon Review Full, loss and accuracy seen in Figures 5 and 4 are similar for

	DBpedia	AG	Sogou	Yelp.p	Yelp.f	Yahoo	Ama.f	Ama.p
Training Accuracy	0.99	0.99	0.93	0.97	0.77	0.74	0.77	0.92
Validation Accuracy	0.99	0.90	0.95	0.97	0.57	0.52	0.72	0.92
Test Accuracy	0.98	0.91	0.95	0.94	0.77	0.72	0.70	0.92

Table 1: Train, Validation, Test Accuracy using DPCNN model.

both datasets, seems to over-fit as the training accuracy increase while the validation accuracy stays constant. These are particularly hard datasets as they are both fine-grained sentiment analysis (5 classes) based datasets with a very diverse population responsible for creating the samples (writing reviews).

5 Analysis

We will analyze our model's predictions on DBpedia test samples, focusing more on incorrectly classified samples.
There are 14 classes in the Dbpedia dataset which are illustrated in Table 2.

The following are the examples of class index predictions by our model:

Class Index	Class
1	Company
2	EducationalInstitution
3	Artist
4	Athlete
5	OfficeHolder
6	MeanOfTransportation
7	Building
8	NaturalPlace
9	Village
10	Animal
11	Plant
12	Album
13	Film
14	WrittenWork

Table 2: Dbpedia Classes

Sample 1: *The James Charnley Residence is located in Chicago's Gold Coast neighborhood in the 1300 block of North Astor Street. The house is now called the Charnley–Persky House and is operated as a museum and organization headquarters by The Society of Architectural Historians (SAH). An Adler Sullivan design the townhouse is the work of Louis Sullivan and a young Frank Lloyd Wright who was a junior draftsman in Sullivan's office at the time.*
Prediction: 7 (Building)
Reference: 7 (Building)
Analysis: The model correctly categorizes the text as "Building". It is able to accurately capture the gist of the sample which talks about a residence. The key words like "residence", "house" and "located" is well taken into account by the model.

Sample 2: *Film Magazine was a film weekly news magazine published in Malayalam Language from Kerala India. It was printed at Thiruvananthapuram and distributed throughout Kerala by Kalakaumudi publications private limited. Even though the magazine had leniages with Kerala Kaumudi news paper its was an independent company. It highlights the doings and happenings of the Mollywood film scene.*
Prediction: 13 (Film)
Reference: 14 (Written Work)
Analysis: This is incorrectly classifies as "film," likely due to the mention of "film" multiple times and "film scene".

Sample 3: *Bulbophyllum mystrophyllum is a species of orchid in the genus Bulbophyllum.*
Prediction: 11 (Plant)
Reference: 11 (Plant)
Analysis: The model correctly recognizes that the text is about a plant though the words "Bulbophyllum mystrophyllum" is not that common, this tells us that our model is able to accurately consider the context such as "species of orchid" in making the predictions.

Sample 4: *Tearoom trade: a study of homosexual encounters in public places is a 1970 book by Laud Humphreys whose Ph.D. dissertation was also titled Tearoom trade. The study is an analysis of homosexual acts taking place in public toilets.*
Prediction: 1 (Company)
Reference: 14 (Written Work)
Analysis: This is incorrectly classified as "com-

pany," likely due to the mention of "Tearoom trade" a couple of times. Here, the model seems to pay more emphasis on a single word and fails to take into consideration the other words like "study" and "analysis" which point to the class "Written Work".

Sample 5: *The Escanaba River is a 52.2-mile-long (84.0 km) river on the Upper Peninsula of the U.S. state of Michigan. In his poem The Song of Hiawatha Henry Wadsworth Longfellow describes how Hiawatha crossed the rushing Esconaba. It is a wide river that cuts into limestone beds. The upper river is rocky and scenic and supports brook brown and some rainbow trout throughout along with warmwater species in the impoundments. John D.*
Prediction: 7 (Building)
Reference: 8 (NaturalPlace)
Analysis: This seems to be a failure case, as there is little context in the document for it to be classified as "building"

Sample 6: *Louise Shivers is an author and writer-in-residence at Georgia Regents University Augusta Georgia.She received a National Endowment for the Arts Fellowship and has published two novels.Here to Get My Baby Out of Jail' originally published by Random House Collins London and Editions Belfond in Paris was named Best First Novel of the Year by USA TODAY in 1983 and was made into the feature film Summer Heat.The movie in now on DVD.The novel is available from John F.*
Prediction: 14 (WrittenWork)
Reference: 3 (Artist)
Analysis: This sample is confusing as it is. Classifying it as "WrittenWork" may not be inaccurate.

Through our sampling, we found very rare cases of failure similar to sample 4. Rest of the incorrectly classified documents seemed to have some context for the predicted incorrect class, and some seemed to be cases where the predicted incorrect class could also be considered an accurate prediction such as sample 5.

6 Conclusion

6.1 Findings

DBpedia has 14 total classes that can be predicted. This large number of classes would be expected to create a high number of categorization errors when compared against a dataset with fewer classes like a polarity dataset. Our findings however show that DBpedia has a high accuracy for the categorization rate. When comparing this result to a dataset like Amazon Reviews polarity, which has only 2 classes, our results suggest that number of classes present is not a the most significant factor in overall model accuracy. When looking at the results of the datasets, a trend occurs where larger datasets perform worse than smaller datasets such as AG News, Sogou News, and DBpedia.

6.2 Improvements

The model can be expanded upon by adding additional repeating blocks, adding depth to the model architecture. Currently we only used 6 repeating blocks, but more of these repeating blocks can be added in the future.

The output layer of the DPCNN can be expanded on by adding a regularizer and weight decay with a parameter of 0.0001 as well as a dropout layer with 0.5. This supposedly will increase the performance according to Johnson and Zhang [5]. These modifications increase training time significantly and therefore, could not have been implemented given our time frame.

6.3 Future Work

The model currently does not finish training when accuracy on the validation set ceases to increase within a reasonable range. The model will continue training even after what appears to be a global maximum is reached. It would be an area for future work to include a stopping mechanism that halts training when the loss no longer decreases more than a certain margin.

References

[1] URL: https://paperswithcode.com/sota/sentiment-analysis-on-amazon-review-full.

[2] URL: https://paperswithcode.com/sota/sentiment-analysis-on-amazon-review-polarityl.

[3] URL: https://pytorch.org/docs/stable/index.html.

[4] Jacob Devlin et al. “BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding”. In: *CoRR* abs/1810.04805 (2018). arXiv: 1810.04805. URL: http://arxiv.org/abs/1810.04805.

[5] Rie Johnson and Tong Zhang. “Deep Pyramid Convolutional Neural Networks for Text Categorization”. In: *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*. Vancouver, Canada: Association for Computational Linguistics, July 2017, pp. 562–570. DOI: 10.18653/v1/P17-1052. URL: https://www.aclweb.org/anthology/P17-1052.

[6] Martin Riedmiller and Heinrich Braun. “A Direct Adaptive Method for Faster Backpropagation Learning: The RPROP Algorithm”. In: *IEEE INTERNATIONAL CONFERENCE ON NEURAL NETWORKS*. 1993, pp. 586–591.

A Accuracy and Loss of Datasets

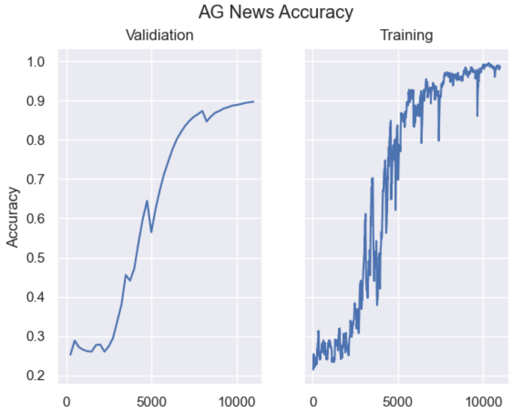

Figure 6: AG News Accuracy

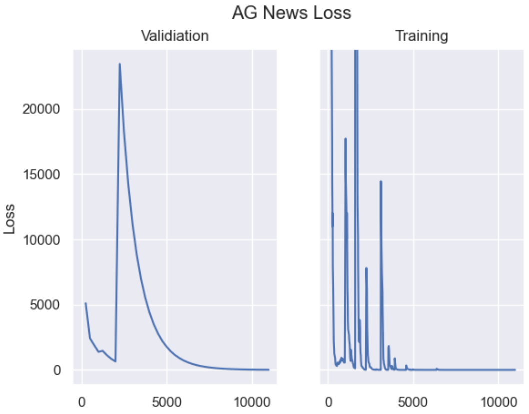

Figure 7: AG News Loss

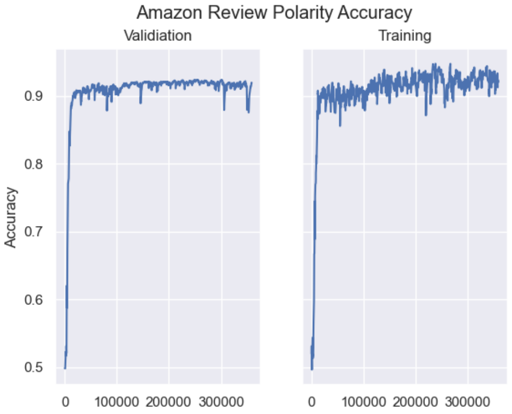

Figure 8: Amazon Review Polarity Accuracy

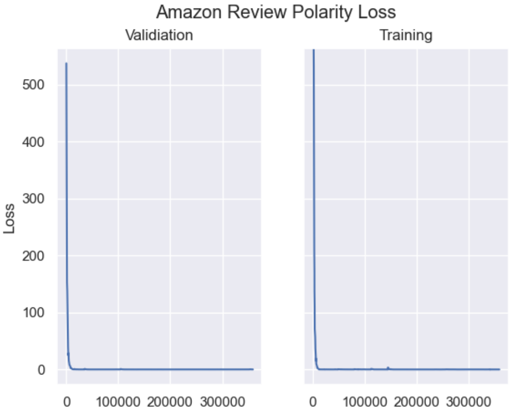

Figure 9: Amazon Review Polarity Loss

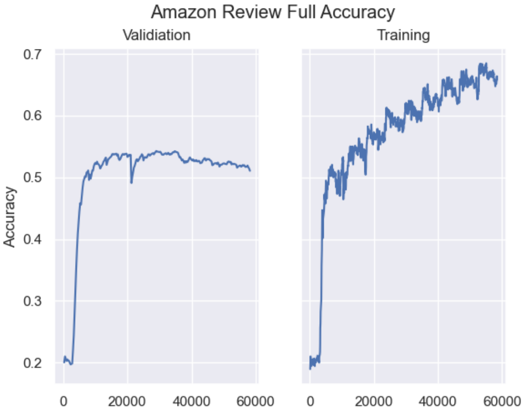

Figure 10: Amazon Review Full Accuracy

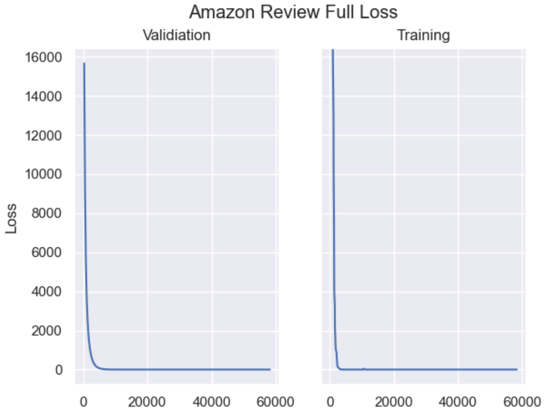

Figure 11: Amazon Review Full Loss

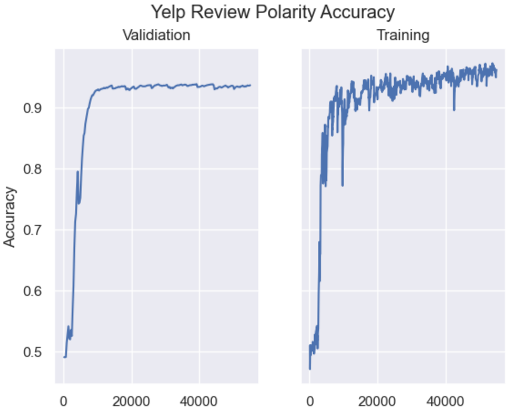

Figure 12: Yelp Review Polarity Accuracy

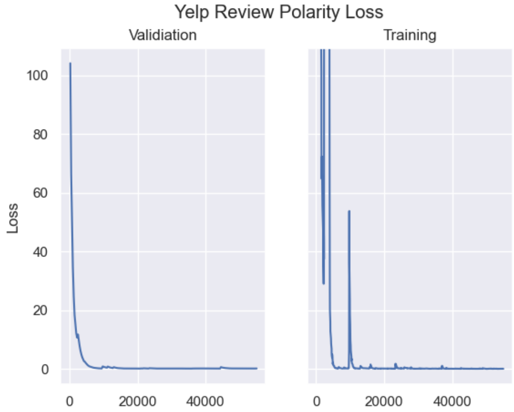

Figure 13: Yelp Review Polarity Loss

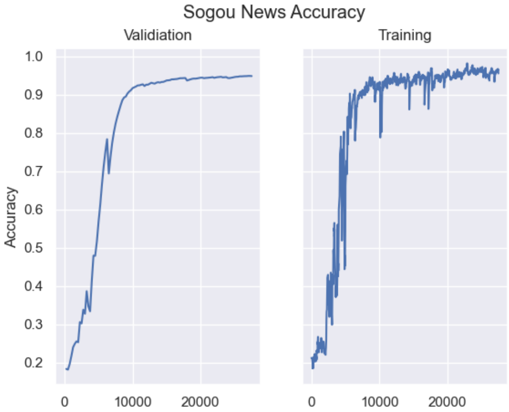

Figure 14: SogouNews Accuracy

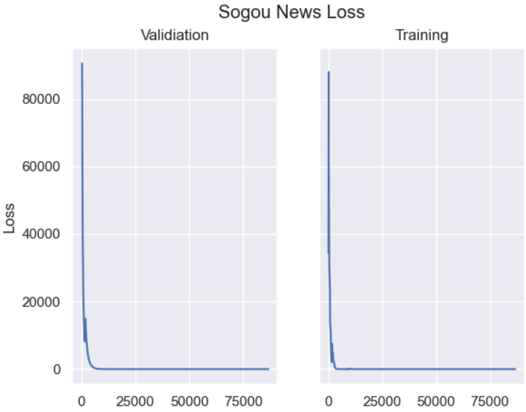

Figure 15: SogouNews Loss