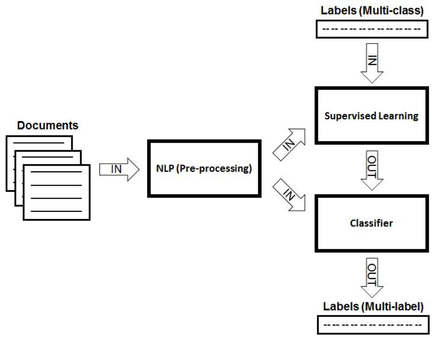
Extracting Tactics from Cyber Security Articles

Jake Smola (smola@stanford.edu)

Abstract

Cybersecurity research is in high supply and security organizations must often commit significant resources to maintaining adequate awareness of new research findings. In this project, we strive to simplify this process by applying text classification to cybersecurity research documents.

We aim to classify cybersecurity documents in a multi-class, multi-label setting. We show that this can be done with an existing implementation called **Magpie**. Magpie is used by the European Organization for Nuclear Research (CERN) and leverages the Word2Vec and Convolutional Neural Network models to perform classification.



Multi-class, Multi-label Classification

Data

Data Source

Our data consists of Palo Alto Networks Unit 42 blog posts. These blog posts are ideal for our task as they tend to be focused on cyber attack TTPs. Additionally, as of the time of this writing, there are over 300 Unit 42 blog posts on record.

Data Labeling

Our labels consist of binary coordinates representing seven cyberattack tactics:

$$\left\{\begin{array}{c}\text{Discovery}\\ \text{Privilege Escalation/Credential Access}\\ \text{Execution}\\ \text{Lateral Movement}\\ \text{Command \& Control}\\ \text{Exfiltration/Collection}\\ \text{Persistence/Defense Evasion}\end{array}\right\} \in \{0,1\}^7$$

As our data did not come pre-labeled, much of our time was spent manually labeling the training data.

Models

Word2Vec

Word2Vec, developed by Mikolov et. al., takes text as input and assigns each word a vector.

The algorithm seeks to maximize the following log likelihood using a **softmax** probability which is computed over all output (word) vectors:

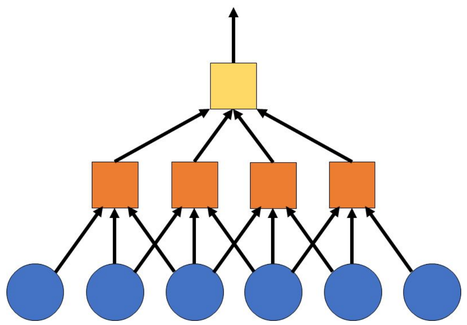
$$\frac{1}{T}\sum_{t=1}^{T}\sum_{-c\le j\le c, j\neq 0}\log p(w_{t+j}|w_t)$$

where $$p(w_O|w_I)=\frac{\exp\left({v'_{w_O}}^{\top}v_{w_I}\right)}{\sum_{w=1}^{W}\exp\left({v'_w}^{\top}v_{w_I}\right)}$$

In doing so, the algorithm can produce vectors that facilitate prediction of surrounding words given a single word as input.

Convolutional Neural Network

A convolutional network is distinct from traditional and recurrent neural networks in that it segregates the inputs into sets such that each first layer unit receives only a subset of the inputs, based on some positional property.

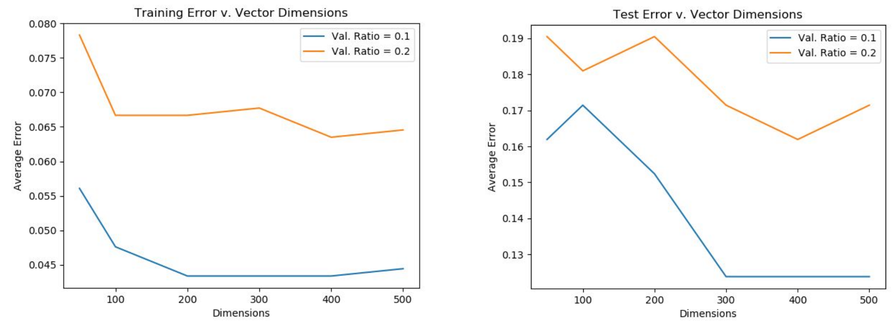


Convolutional Neural Network (CNN)

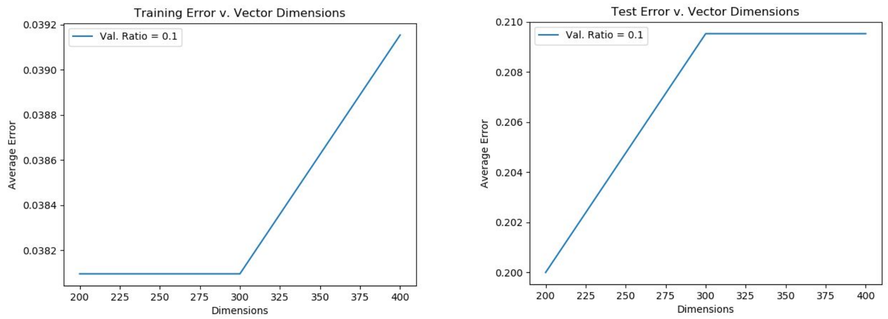
The convolutional neural network within Magpie passes chunks of contiguous word vectors to the first layer; subsequent layers compute softmax activation functions before the final layer aggregates and outputs the prediction.

Experiments

We train Magpie's classifier using three, random permutations of 50 samples. In our first experiment, we consider two training-validation-test splits: 40-5-5 and 36-9-5; as well as five word2vec dimension parameters: increments of 100 from 100 to 500. For each combination of data split and dimension parameter, we train Magpie for 100 epochs on each of the three data permutations and average the training and test errors. We then adjust our previous parameters based on the resultant error and rerun the experiment using only the 40-5-5 split and dimensions 200, 300, and 400 over 150 epochs.



Experiment 1 Results



Experiment 2 Results

Discussion

Training Error Analysis

Given the scope of our problem and limited dataset at our disposal, the training error obtained during both experiments across all iterations shows that the Magpie implementation is quite successful maximizing the objective function. Epoch progress shows substantial improvements in loss for each trial and there appears to be a training "sweet-spot" between 200 and 400 dimensions for both splits. Experiment 1 witnessed a fairly higher training error for the 36-9-5 split, most likely due to fewer training samples. Experiment 2 incurred even lower training error when given 50 more epochs to run. Nonetheless, we believe more training samples will reduce the training error and improve overall prediction accuracy, since our full label space cannot be not fully represented by the 50 training samples alone.

Test Error Analysis

Overall, the test error achieved in both experiments is rather acceptable for the problem we are trying to solve. As in the case of training error, experiment 1 saw substantially lower test error from the 40-5-5 split and in higher dimensions. While experiment 2 was able to achieve improved training error results, it incurred higher test error across all tested dimensions. This result suggests the experiment's additional epochs may have caused an overfit in training.

Future

Data Acquisition

Future work would focus on acquiring additional labeled data to facilitate improved training quality and ultimately to support higher test accuracy.

Additional Tuning

Although we were able to explore a variety of parameters, our analysis was limited to a preselected subset. Future iterations of this work could focus on validating the effects of additional parameters on training and test error as well allowing greater flexibility in the values assigned to each parameter.

Regularization and Early Stopping

Future iterations can also implement regularization and early stopping, as this will likely reduce overfitting to the parameters of the model and result in improved test accuracy.

References

Adversarial Tactics, Techniques \& Common Knowledge. MITRE. Available from: https://attack.mitre.org/wiki/Main_Page

Batista, D. S. Document Classification. Available from: http://www.davidsbatista.net/blog/2017/04/01/document_classification/

Blei, D., Ng, A., and Jordan, M. Latent Dirichlet allocation. Journal of Machine Learning Research, 3:993-1022, 2003. Available from: http://ai.stanford.edu/~ang/papers/nips01-lda.pdf

Chen, S., Soni, A., Pappu, A., and Mehdad, Y. DocTag2Vec: An Embedding Based Multi-label Learning Approach for Document Tagging. Available from: https://arxiv.org/abs/1707.04596

Mikolov, T. Sutskever, I., Chen, K., Corrado, G., Dean, J. Distributed Representations of Words and Phrases and their Compositionality. Computation and Language. Available from: https://arxiv.org/abs/1310.4546

Natural Language Toolkit. Available from: http://www.nltk.org/

Stypka, J. Magpie. Github. Available from: https://github.com/inspirehep/magpie

Unit 42. Palo Alto Networks. Available from: https://researchcenter.paloaltonetworks.com/unit42/

Wang, C. Supervised latent Dirichlet allocation for classification. Available from: http://www.cs.cmu.edu/~chongw/slda/