Article Theme Classification Based on Titles
Milestone Report II

Problem Statement

See Milestone Report I.

Data Source

See Milestone Report I.

Data Cleaning

See Milestone Report I.

Exploratory Analysis

See Milestone Report I.

Recurrent Network

I experimented on recurrent neural networks of different configurations. Vector representations generated by bag-of-words models are not suitable as input to a recurrent network as bag-of-words representations are not sequentially dynamic. I used the continuous bag-of-words model to generate word2vec embeddings. The continuous bag-of-words method trains a network that can predict the target word given the surrounding contextual words. Word2vec projects each word into a high-dimension vector space where contextually close words correspond to vectors located spatially close.

Individual words, which occur more than (including) twice in the training set, were mapped to 128-dimension vectors as the algorithm utilized the 6 words around it in the text as the context. 128 dimensions and 6-word-wide context were chosen after experimenting with various configurations. Higher dimensions and more lengthy context did not improve the performance of the classifiers. The word2vec model was trained on the training set solely. I then converted the titles in both the training and test sets to sequences of vectors using the word2vec model. Novel words that were not included in the vocabulary formed during training were mapped to a vector with all elements equal to 0. As optimization algorithms for the neural network require input sequences to be of uniform length, all sequences were pre-padded with 0-vectores.

After testing multiple architectural configurations, a network with two recurrent layers followed by two perceptron layers demonstrated best performance. In each layer, there were 256 units. The network achieved an accuracy of 22%, and an average multiclass precision of 22%. Consistent with its worse accuracy compared to the naïve Bayes classifier, the model presented an inferior precision-recall curve – f1 score less than 0.2 (Figure 1). To reach a 90% average multiclass precision, the minimum threshold should be 0.984; the corresponding average

multiclass recall is 0.6%. The confusion matrix (Figure 2) demonstrates similar misclassification patterns to that of the naïve Bayes model and the multilayer perceptron network.

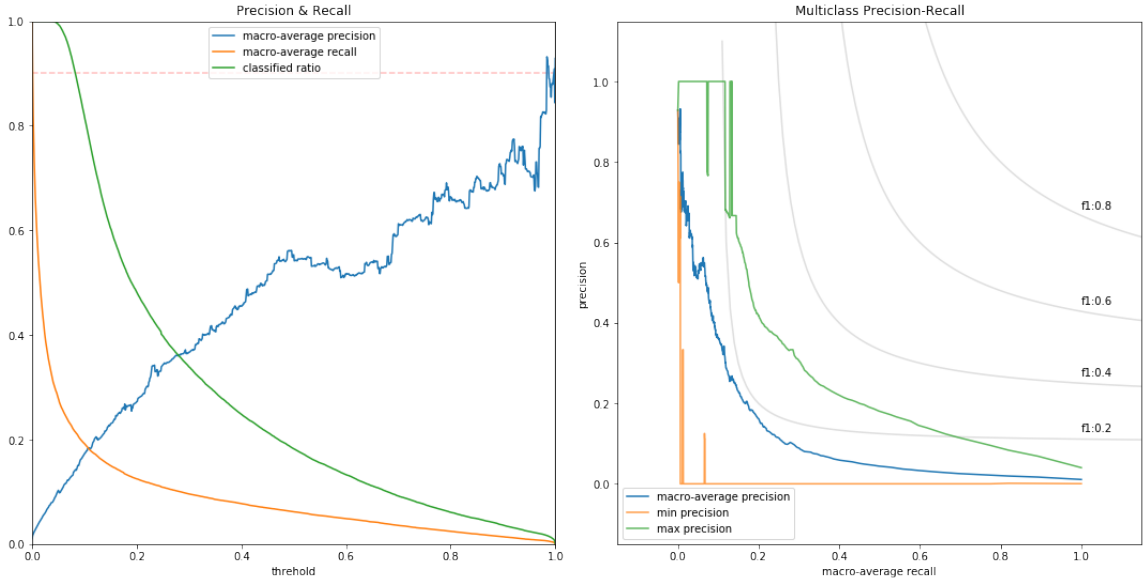

Figure 1. Precision-recall of recurrent network

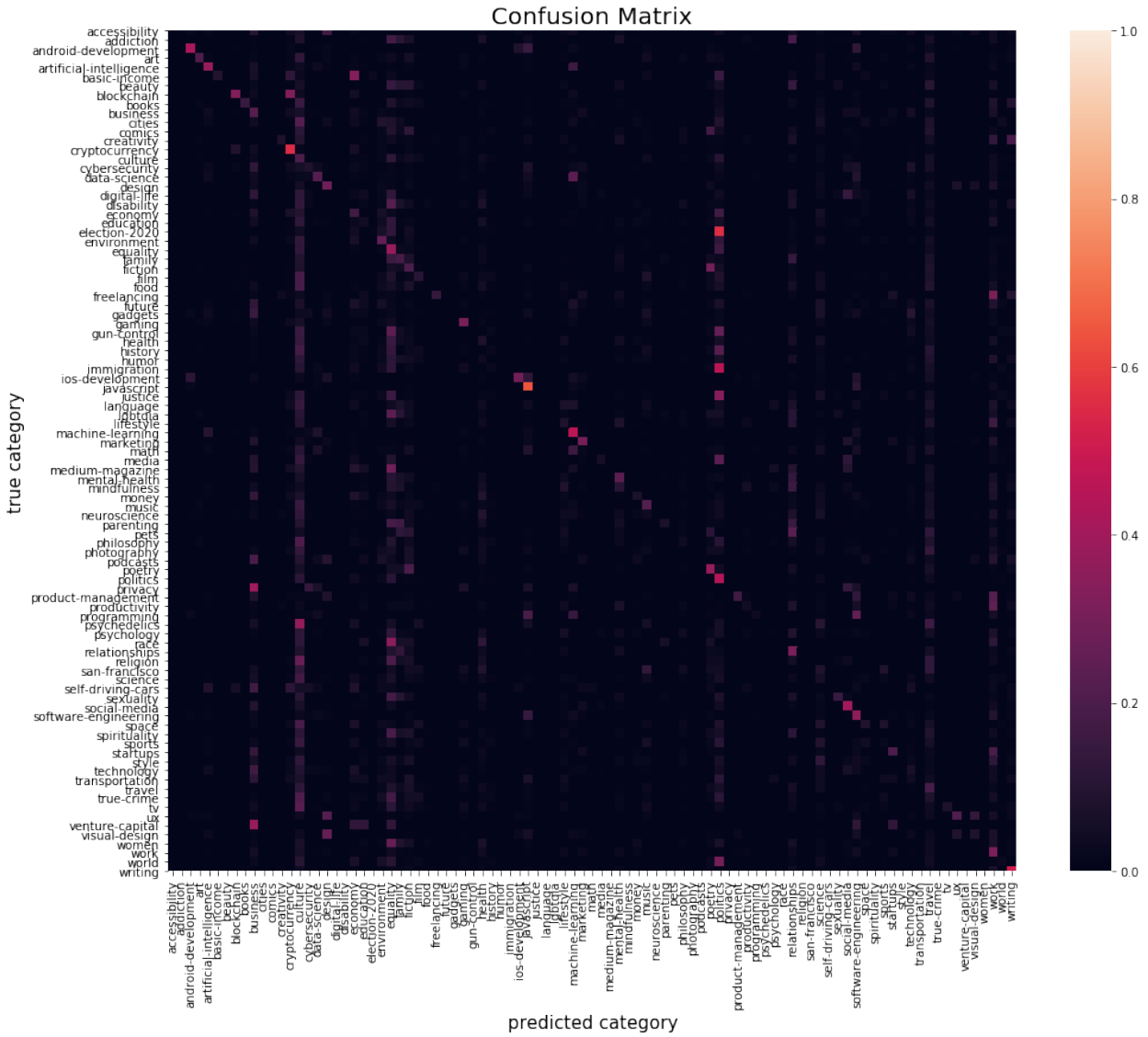

Figure 2. Precision-recall of multilayer perceptron network

Long Short-Term Memory Network

The long short-term memory network also takes input of continuous bag-of-words embeddings. For the recurrent network, the word2vec model converted each word to a 128-dimension vectors, and increasing the dimensionality of the vector representation did not improve the recurrent network's performance. However, increasing the dimensionality of the input was able to boost the accuracy of the LSTM network. Thus, for the LSTM networks, each word was mapped a 512-dimension vector.

After varying the configuration of the LSTM network, it was observed that with the limited computational resources I have access to, a network with a layer of 512 LSTM units followed by a 512-perceptron layer gave the most cost-effective results. It achieved about 29% accuracy on the test set, and 26% average multiclass precision.

Compared to the recurrent network, the LSTM classifier performed better in terms of accuracy and its precision-recall curve reached better f1-score level, although not as good as the multilayer perceptron network or the naïve Bayes classifier. However, the increased accuracy is reflected more in the recall while the precision is significantly inferior to the other three models (Figure 3). The confusion matrix (Figure 4) shows that the LSTM network suffered very similar misclassification error to that of the other three models.

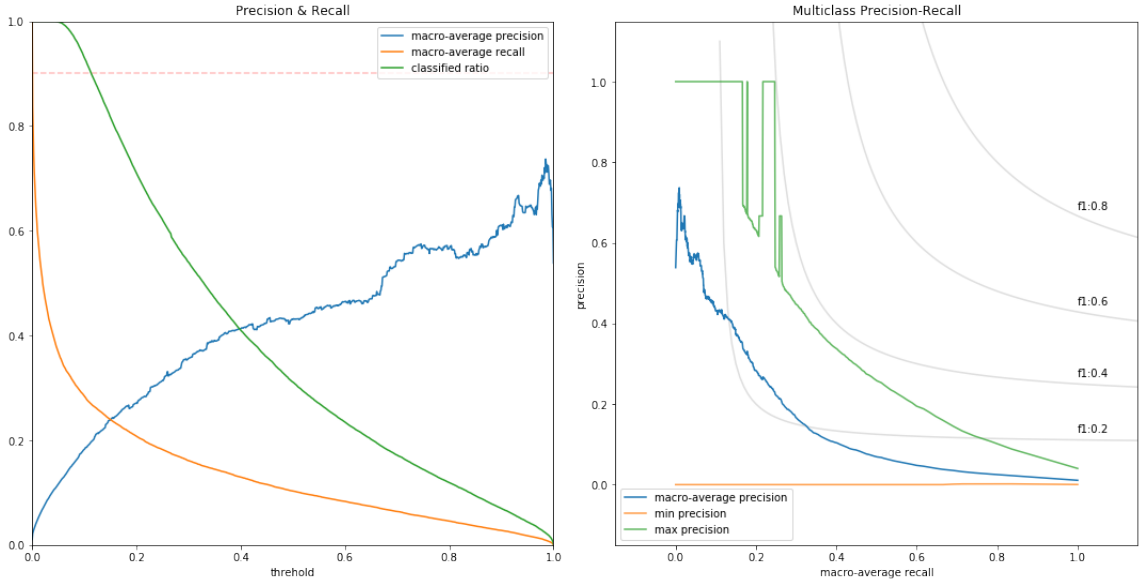

Figure 3. Precision-recall of LSTM network

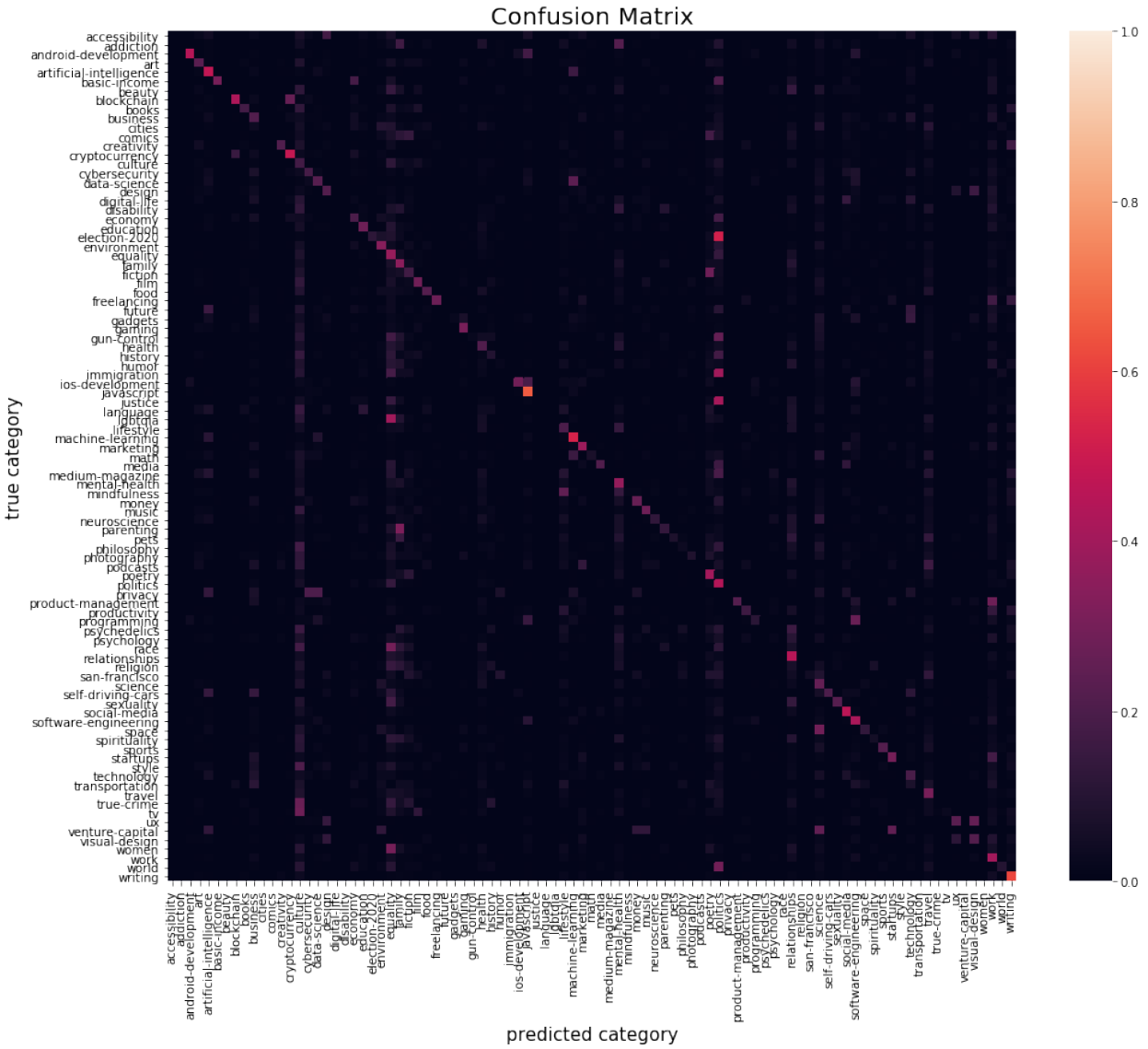

Figure 4. Confusion matrix of LSTM network

Comparing Models

In terms of accuracy, the naïve Bayes and multilayer perceptron network models perform significantly better than the recurrent network and the LSTM network. Similarly, at the same probability threshold, the naïve Bayes and multilayer perceptron network classifiers show significantly better precision (Figure 5).

The reason behind the inferior performance of the recurrent network and the LSTM network might be two folds – the type of vectorization method they require and the mismatch between the characteristics of the data and the specialty of the networks. Both the naïve Bayes classifier and the multilayer perceptron network take input of bag-of-words vector representations. These vectors are very sparse and static, and the models are ignorant of the words' meaning, which can be ideal for short texts. On the other hand, the recurrent network and the LSTM network accept

input from continuous bag-of-words embeddings. The model was trained to be aware of the words' meaning based on context, and the vector representations are dynamic and sequential. On the surface, the recurrent network and the LSTM network may be expected to take advantage of the more sophisticated input as it is more meaningful and dynamic. However, in this case, the classification is based on only titles, which tend to be not lengthy, thus maybe providing insufficient context to build a strong word2vec model.

The recurrent network and the LSTM network process sequential data, and the input sequence should be long enough, especially for the LSTM network, for the network to learn any meaningful dynamic patterns from the data. However, as titles are usually very concise, a lot of them may not be suitable for recurrent and LSTM networks.

Between the naïve Bayes model and the multilayer perceptron network, the naïve Bayes model demonstrates better precision, but the multilayer perceptron network excels in terms of recall and the number of titles that got classified (Figure 5).

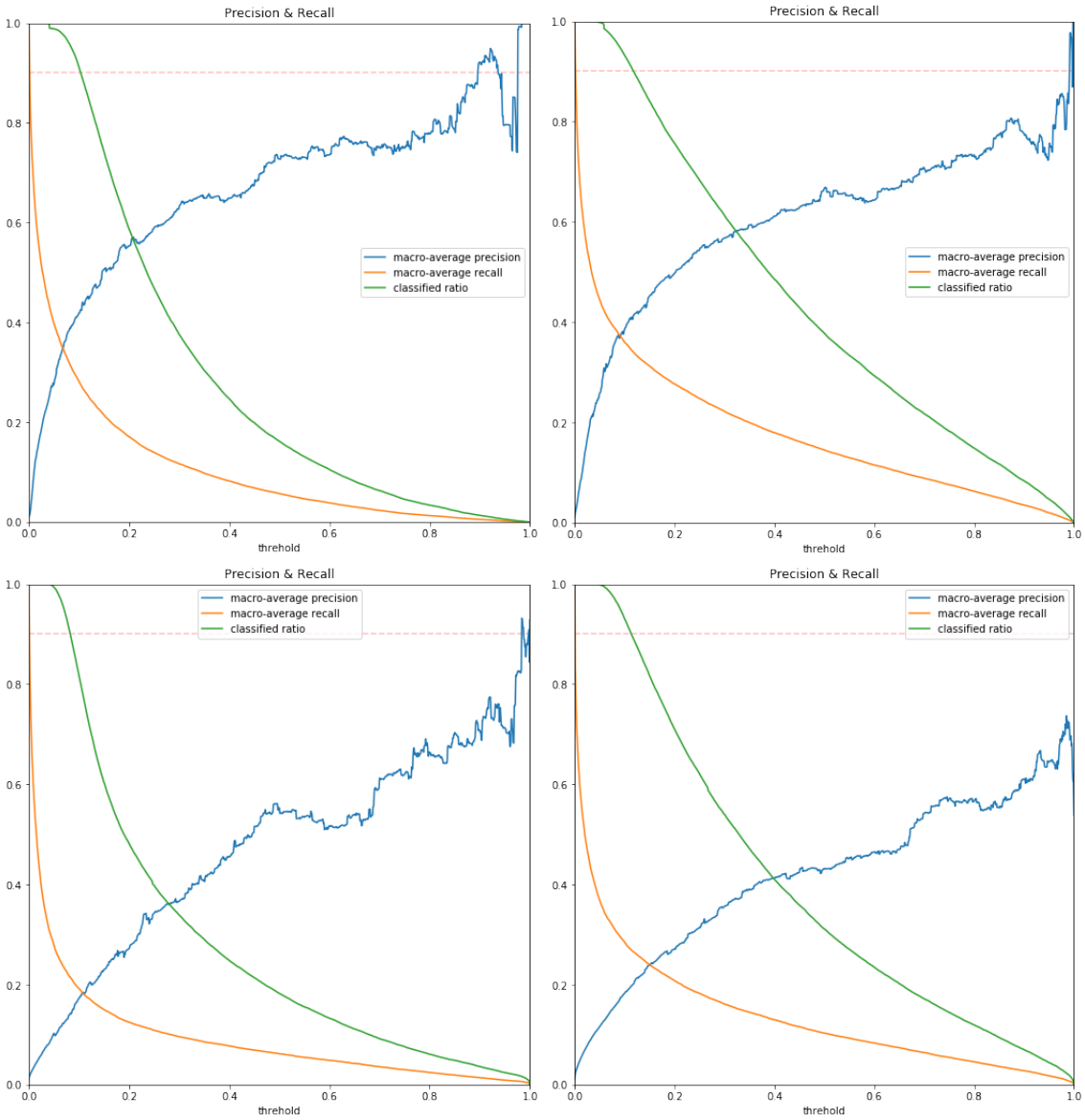

Figure 5. Comparing precision and recall of different models – naïve Bayes (upper left), multilayer perceptron network (upper right), recurrent network (lower left) and LSTM network (lower right)

If a recommendation system is to be built, where given a probability threshold for the chosen classifier, all themes with predicted probability higher than the threshold should be suggested as possible choice for the user to choose from, a correct suggestion rate can be calculated for the proportion of documents whose suggested themes include the real theme. In this sense, for a very low threshold, the correct suggestion rate would only be slightly higher for the naïve Bayes classifier (although the recommendations would also include a lot of inaccurate and redundant themes). On the other hand, and more importantly, for higher and more reasonable working thresholds, the correct suggestion rate is significantly higher for the multilayer perceptron network (Figure 6). This suggests the multilayer perceptron network may be a better model to use when building such recommendation systems.

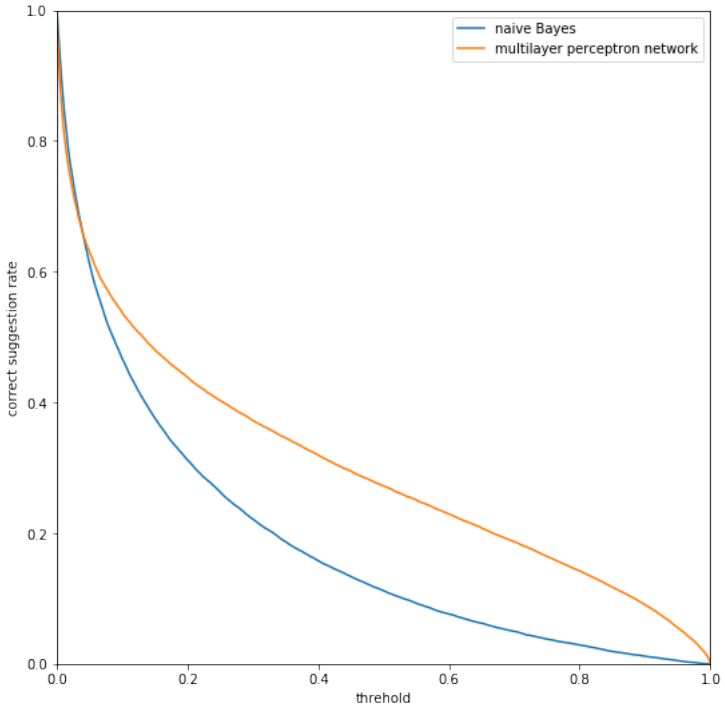

Figure 6. Correct suggestion rate of naïve Bayes model and multilayer perceptron network