Chapter 17
Intra-Domain Text Classification: A Hybrid Approach

Soumak Chakraborty, Himadri Mukherjee, and Alo Ghosh

Abstract The amount of textual information has been increasing at an enormous rate in the digital world. This has led to the development of efficient indexing mechanisms for easier retrieval. One of the primal attributes for categorizing texts is based on their domain. This is a challenging affair due to the commonality of vocabulary. The challenge further aggravates during deeper sub-domain classification. This very important as the rustics (especially students) often need information which concerns a particular subject. Systems capable of organizing information based on subjects can tremendously aid towards efficient retrieval in these scenarios. In this paper, a system is presented to classify educational documents amidst three subjects: computer science, physics and mathematics. Experiments were performed with over 13K research papers, and the highest accuracy of 93.35% was obtained for intra-domain classification using a hybrid technique comprising both handcrafted features and deep learning.

17.1 Introduction

There has been a large increase in the amount of textual information in the digital world. This has been accompanied by a tremendous number of accesses as well. To ensure efficient retrieval of such information, proper indexing is critical. One of the most common approaches of indexing textual information is based on domain. However, this is not adequate in disparate scenarios, especially for educational

The authors contributed equally

S. Chakraborty (✉) · H. Mukherjee · A. Ghosh
AILabs, Kolkata, West Bengal, India
e-mail: soumak.chakraborty@ailabs.academy

H. Mukherjee
e-mail: himadri.mukherjee@ailabs.academy

A. Ghosh
e-mail: alo.ghosh@ailabs.academy

V. Bhateja et al. (eds.), *Evolution in Computational Intelligence*, Smart Innovation, Systems and Technologies 326, https://doi.org/10.1007/978-981-19-7513-4_17

documents. The education domain comprises of multiple subjects and students often search information concerning a particular subject. This sets the need for systems which can categorize educational information with respect to subjects. This will enable easier and more efficient access of information. Performing intra-domain classification is a challenging task. This is mostly due to the overlap of not only words but phrases as well.

Dhar et al. [1] have discussed disparate techniques of text categorization. Parida et al. [2] performed text categorization on the Reuters-21758 dataset. The texts were parameterized using TF-IDF features, and thereafter Chi-square test was used to select the best 1000 features. Random forest and naive Bayes were used for classification, and better performance was reported for naïve Bayes. Dhar et al. [3] performed text categorization of Bangla news text. The dataset consisted of nine domains, namely Business, Entertainment, Food, Literature, Medical, State Affairs, Sports, Science and Technology, and Travel. The texts were parameterized using graph-based feature which was followed by LSTM-based classification and the highest accuracy of 99.21% was reported. Hao et al. [4] distinguished eight different domains from 400 pages of web text data. The considered domains were Finance, IT, Health, Sport, Travel, Education, Culture and Military. The texts were modelled using TF-IDF features along with naïve Bayes and SVM, wherein better performance was obtained for naïve Bayes. Xue and Li [5] presented a system to categorize texts from the Sogou laboratory text categorization corpus. They used the bigram method for word segmentation followed by random forest Classifier. They performed experiments for different tree and feature sizes, wherein the best result was reported for tree and feature size of 250 and 900, respectively.

Kibriya et al. [6] performed text categorization using disparate datasets including 20 newsgroups WebKB and Reuters-21578 Standard and normalized TF-IDF features were extracted post-bag-of-words technique. Different varieties of naïve Bayes classifier were explored along with SVM, wherein the best result was obtained using SVM. Vaissnave and Deepalakshmi [7] attempted text categorization from Indian legal documents. The classes included Fact, Issue, Arguments of petitioner, Arguments of responder, Reasoning, Decision, Majority concurring and Minority dissenting. The texts were vectorized using Word2Vec technique, and an accuracy of 88% was reported using bidirectional LSTM-based classification. Lade and Dhore [8] attempted to classify Marathi texts amidst three categories, namely sports, entertainment and economy. The texts were parameterized using TF-IDF features and then fed to a KNN-based classifier which yielded the highest accuracy of 91.27%. Ahmed et al. [9] categorized Bangla texts into ten categories. They experimented with attention-based RNNs and BiLSTM which fetched acccuracies of 97.72% and 86.56%, respectively. Bahassine et al. [10] presented an improved version of Chi-square feature selection technique for categorizing Arabic texts. Experiments were performed with 5070 documents spanning over 6 domains, and the best F-score of 90.50% was reported with 900-dimensional features.

It is observed that most of the works concentrate on broad domain categorization and developments for intra-domain categorization has been on the lower side. This is an important aspect for faster retrieval of documents considering the fact that every domain encompasses a vast variety of information.

17.2 Proposed Method

In the current experiment, two different approaches were employed. In the former, a deep learning-based technique was used on the raw data. In the other part, the articles were parameterized using handcrafted features which were then supplied to a deep learning-based classifier. The details are presented in the subsequent paragraphs.

17.2.1 Deep Learning-Based Approach

In this technique, the words were vectorized to a numerical format which were then fed to a LSTM-based classifier which is detailed hereafter.

17.2.1.1 Word Embedding

The words for each of the articles were embedded/vectorized using Word2Vec [11] technique. These produced vector representations for every word which was constructed by considering the part of speech, disambiguated sense, syntax and semantics of the text. For every word, a 300-dimensional vector was obtained. Each of these vectors was amalgamated to form the entire text block. As different instances, had disparate word counts, so feature vectors of multifarious dimensions were obtained. In order to vectors of constant dimension, the amalgamated vectors were zero-padded to the length of the largest vector. The vectorized representation of 30 different words is presented in Fig. 17.1.

17.2.1.2 Long Short-Term Memory Network

LSTM or long short-term memory network [12] is an improvement over the standard recurrent neural network which solves the problem of remembering long context in text due to its memory capability and the problem of vanishing gradient that is evident in recurrent neural networks. The LSTM network is composed of multiple cells along with forget, input and output gates. The forget gate is responsible for discarding information and takes previous hidden cell state as the forget gate which works with the previous hidden state and the present input. This is followed by the input gate which adds information to the cell state. It uses a sigmoidal function to

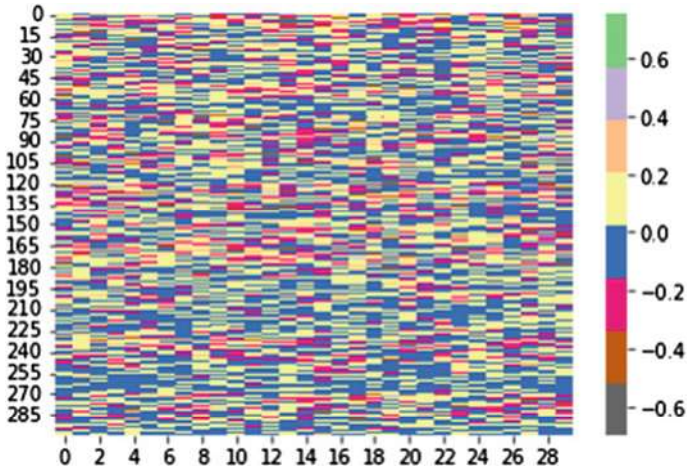

Fig. 17.1 Embedded representation of 30 words from the dataset

regulate the added values. This is finally followed by the output gate which selects meaningful information from the current cell state. This information is provided as output. The LSTM layer is followed by dense layers which are fully connected layers. They perform classification based on the input from the previous layers. In this experiment, the network comprised of a single LSTM layer with a default dimension of 50. Thereafter, a dropout layer was introduced with a 50% parameter discarding scheme (default) to prevent overfitting. This was followed by two dense layers. The 1st dense layer had a default dimension of 50 and the final output layer had a dimension of 3. The LSTM layer had tanh activation while the dense layers had ReLU and softmax activations which are presented in Eqs. (17.1) and (17.2), respectively. The number of generated parameters for the best network is presented in Table 17.1.

$$f(x) = \max(0, x), \tag{17.1}$$

here, x is the input to a neuron.

Table 17.1 Number of generated parameters for the proposed network

Layer	Parameters
Embedding	1500000
LSTM	70200
Dense 1	6375
Dense 2 (output)	378
Total	1576953

$$\sigma(z)_j = \frac{e^{z_j}}{\sum_{k=1}^{K} e^{z_k}}, \tag{17.2}$$

where z is an input vector of length K.

17.2.2 Hybrid Approach

In this technique, term frequency–inverse document frequency (TF-IDF) features [13] were extracted for the words at the outset. The TF-IDF feature is composed of two aspects which are discussed as follows:

- Term Frequency (TF): Number of times a word is present in a sentence/total number of words in the sentence.
- Inverse Document Frequency (IDF): log(Total number of sentences/Number of sentences the specific word appears)

Finally, the TF-IDF feature is obtained by multiplying TF and IDF matrices. The maximal feature dimension was set to 5000 after trial. The words were converted to lowercase for avoiding duplicate entry due to capitalization, and only unigrams were considered. Stop words were removed using a standard list of English stopwords comprising of 318 entries. The extracted features were thereafter fed to the LSTM network.

17.3 Results and Discussion

The experiments were performed in two phases whose results are presented hereafter. The first phase presents the results for the deep learning-based approach. This phase was used to tune the LSTM network as well, and a 80:20 train test split was used. The second phase presents the outcome of the hybrid approach, and to test the robustness of the system, fivefold cross-validation was used.

17.3.1 Dataset

Data is a very important aspect for any experiment. The quality of data plays an important role in judging the robustness of a proposed system. It is important for the dataset to uphold real-world characteristics. In this experiment, a dataset of abstracts from research articles was used.[1] There are five different subjects, namely physics,

[1] https://www.kaggle.com/datasets/vetrirah/janatahack-independence-day-2020-ml-hackathon.

mathematics, computer science, statistics, quantitative finance and quantitative biology. Out of them the 1st were considered to class imbalance which was observed for the other subjects. Moreover, these three subjects are very much related to one another and at times have same keywords as well. There were 4910, 5120 and 3610 instances for physics, computer science and maths, respectively, totalling to 50412 unique tokens.

17.3.2 Deep Learning-Based Technique

Initially, the dimension of the LSTM layer was varied from 25 to 125 with a step of 25 whose results are presented in Table 17.2. The best performance was obtained for 50-dimensional LSTM. The dimension of the dense layer was set to 50 (default).

The dimension of the dense layer was varied from 25 to 150 with a step of 25. The results are presented in Table 17.3. The LSTM dimension was set to the default value of 75. It is noted that the best performance was obtained for 100-dimensional dense layer.

Finally, the best LSTM network comprising of 50-dimensional LSTM layer accompanied by 50% dropout followed by 100 and three-dimensional dense layers was obtained which was trained using disparate batch sizes whose performances are tabulated in Table 17.4. It is noted that the best performance was obtained for a batch size of 64 instances.

The best-performing setup was thereafter used to evaluate the system on the entire dataset using fivefold cross-validation where an accuracy of 90.32% was obtained. This was done to test the robustness of the proposed system for disparate train–test combinations and also to ensure that every instance in the dataset was subjected to test set atleast once.

Table 17.2 Performance for different LSTM dimensions

LSTM dimension	25	50	75	100	125
Accuracy	90.61	91.42	88.56	89.52	90.76

Table 17.3 Performance for different dimensions of the intermediate dense layer

Dense dimension	25	50	75	100	125	150
Accuracy	89.59	90.14	90.18	90.47	91.50	91.02

Table 17.4 Performance for different batch sizes during training

Batch size	8	16	32	64	128
Accuracy (%)	89.66	89.88	91.61	91.24	90.54

Table 17.5 Confusion matrix for the hybrid system

	Physics	Computer	Maths
Physics	4597	115	198
Computer	159	4853	108
Maths	206	121	3283

Table 17.6 Performance of TF-IDF features

Classifier	SVM	RF	MLP	NB	LSTM
Accuracy (%)	93.27	90.32	92.61	91.77	93.35

17.3.3 Hybrid Technique

The handcrafted TF-IDF features were fed to LSTM network, and an accuracy of 93.36% was obtained using cross-validation whose interclass confusions are presented in Table 17.5. It is noted that the highest confused pair was physics and maths. One of the probable reasons for this is the similarity of terminology amidst the two subjects. The confusions for computer science with the other two subjects were almost 43% lower than that of physics and maths. One of the primal reasons for this was the marginally higher number of data for computer science. Further analysis also revealed lower similarity of computer science texts with the other two subjects in disparate cases which possibly led to lower confusion.

The TF-IDF features were tested with other popular classifiers in the thick of SVM [14], random forest (RF) [15], multilayered perceptron (MLP) [16] and naive Bayes (NB) [17]-based classifiers. The results are presented in Table 17.6. It is noted that LSTM produced the best average performance amidst all the classifiers which was followed by SVM. The lowest performance was obtained for random forest. The foldwise performance is presented in Fig. 17.2. It is seen that LSTM consistently

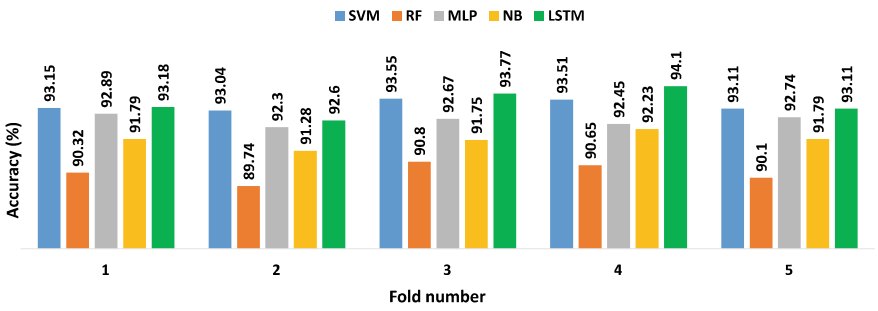

Fig. 17.2 Performance of the classifiers in every fold

produced the best performance for every fold except for fold 2 where SVM performed better. In the last fold, the performance of SVM and LSTM was the same. In all the folds, MLP ranked 3rd followed by naive Bayes and random forest.

17.4 Conclusion

In this paper, a hybrid technique of intra-domain text classification is presented. The system works with TF-IDF features and LSTM-based classification. The system was tested with abstracts of research articles from three different subjects, wherein a weighted precision of 0.93 was obtained. In future, we will test the system with more subjects from single and multiple domains. We also plan to parameterize the texts with other handcrafted features. Usage of other embedding techniques as well as deeper networks is also in our future plans. Finally, the system will be trained using data augmentation for better performance and will be deployed in the web to test its performance for real-time text categorization.

References

1. Dhar, A., Mukherjee, H., Dash, N.S., Roy, K.: Text categorization: past and present. Artif. Intell. Rev. **54**(4), 3007–3054 (2021)
2. Parida, U., Nayak, M., Nayak, A.K.: News Text Categorization using random forest and Naïve Bayes. In: 2021 1st Odisha International Conference on Electrical Power Engineering, Communication and Computing Technology (ODICON), pp. 1–4. IEEE (2021)
3. Dhar, A., Mukherjee, H., Obaidullah, S., Santosh, K.C., Dash, N.S., Roy, K.: Web text categorization: a LSTM-RNN approach. In: International Conference on Intelligent Computing and Communication, pp. 281–290. Springer, Singapore (2019)
4. Hao, P., Ying, D., Longyuan, T.: Application for web text categorization based on support vector machine. In: 2009 International Forum on Computer Science-Technology and Applications, vol. 2, pp. 42–45. IEEE (2009)
5. Xue, D., Li, F.: Research of text categorization model based on random forests. In: 2015 IEEE international conference on computational intelligence and communication technology, pp. 173–176. IEEE (2015)
6. Kibriya, A.M., Frank, E., Pfahringer, B., Holmes, G.: Multinomial naive bayes for text categorization revisited. In: Australasian Joint Conference on Artificial Intelligence, pp. 488–499. Springer, Berlin, Heidelberg (2004)
7. Vaissnave, V., Deepalakshmi, P.: A keyword-based multi-label text categorization in the Indian legal domain using Bi-LSTM. In: Soft Computing: Theories and Applications, pp. 213–227. Springer, Singapore (2022)
8. Lade, S., Dhore, M.L.: Text categorization of Marathi news articles using machine learning. In: Proceeding of First Doctoral Symposium on Natural Computing Research, pp. 63–72. Springer, Singapore (2021)
9. Ahmed, M., Chakraborty, P., Choudhury, T.: Bangla document categorization using deep RNN model with attention mechanism. In: Cyber Intelligence and Information Retrieval, pp. 137–147. Springer, Singapore (2022)

10. Bahassine, S., Madani, A., Al-Sarem, M., Kissi, M.: Feature selection using an improved Chi-square for Arabic text classification. J. King Saud Univ. Comput. Inf. Sci. **32**(2), 225–231 (2020)
11. Church, K.W.: Word2Vec. Natural Lang. Eng. **23**(1), 155–162 (2017)
12. Yu, Y., Si, X., Hu, C., Zhang, J.: A review of recurrent neural networks: LSTM cells and network architectures. Neural Comput. **31**(7), 1235–1270 (2019)
13. Aizawa, A.: An information-theoretic perspective of TF-IDF measures. Inf. Process. Manage. **39**(1), 45–65 (2003)
14. Pisner, D.A., Schnyer, D.M.: Support vector machine. In: Machine learning, pp. 101–121. Academic Press (2020)
15. Breiman, L.: Random forests. Mach. Learn. **45**(1), 5–32 (2001)
16. Gardner, M.W., Dorling, S.R.: Artificial neural networks (the multilayer perceptron)-a review of applications in the atmospheric sciences. Atmos. Environ. **32**(14–15), 2627–2636 (1998)
17. Webb, G.I., Keogh, E., Miikkulainen, R.: Naïve Bayes. Encyclop. Mach. Learn. **15**, 713–714 (2010)