CATEGORIZATION OF TEXT WITH RNN, CNN AND HAN MODELS

Ridhima Dhawan

School of Computer Science and Engineering,

Lovely Professional University,

Phagwara, Punjab, India.

ABSTRACT

In this paper, few distinctive machine learning algorithm strategies recurrent neural network (RNN), convolutional neural network (CNN) and hierarchical attention network has been used to classify the text and performances has been evaluated. Classification of text was executed on datasets having Frennch, Italian, Germman, and Englishh languages. The execution is entirely based on the Kera's. It characterizes the time taken by each machine learning algorithm RNN, CNN, HAN for processing the text data and also, it epitomizes the training accuracy and validation accuracy of each machine algorithm on the dataset1, dataset2, dataset3 respectively.

Keywords: Text Classification, RNN, CNN, HAN, Kera's

I. INTRODUCTION

Text classification is a process of assigning categories to text or categorising text into different classes or labels using Natural language processing and Supervised Machine Learning (ML). It is an example of Supervised Machine Learning mission since a labelled dataset containing text files and their labels is used for training a classifier. The objective of text category is to categorize the textual content files into one or more predefined categories. Its wide-ranging applications include: sentiment analysis from social media, junk mail recognition & no unsolicited emails, tagging of customer enquiries automatically, Classification of news and articles into predefined subjects.

The researcher chooses from best features to best ml classifiers for classification of the text. All the techniques of text classification grounded on the words.

In this paper, few distinctive machine learning algorithm strategies recurrent neural network (RNN), convolutional neural network (CNN) and hierarchical attention network (HAN) [1, 2]has been used to classify the text and performances has been evaluated. The implementation is completely based on the Kera's. Classification of text was performed on datasets having French, Italian, German, and English languages.

II. RELATED WORK

In the research, text classification model is consisting of training text, feature vector, labels machine learning algorithm and predictive model mechanisms. The input is provided to the training text, in the form of text through which our superrvised learnning modell is able to acquire and forecast the vital class. A vector of feature is a vector that covers data describing the features of the input data. Labels are the predefined categories/classes that our model would suppose**.** CNN RNN HAN, machine learning algorithm, is used to deal with text classification. Predictive Model**,** a label prediction is performed on the basis of the historical dataset.

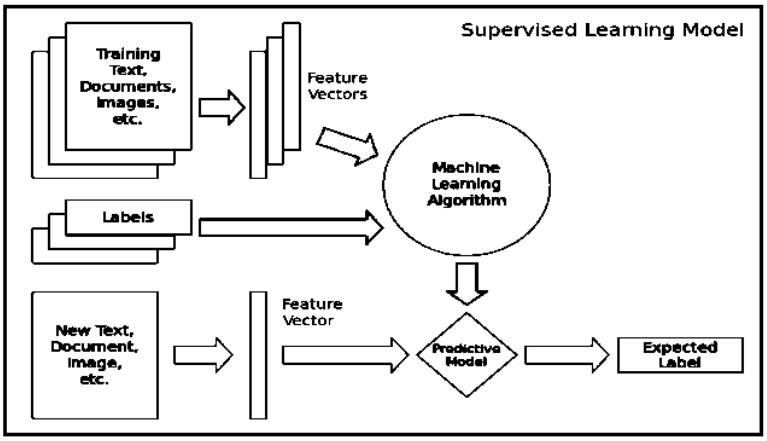

Fig 1. the above picture represents the supervised learning model adopted for the text classification

A. Analysing Data:

The three categories of dataset with numerous classes as shown in table below:

DATASET	DATA SIZE	CLASS	TRAIN SAMPLES
DATASET1	17325	350	14658/3664
DATASET2	3155	29	2524/631
DATASET3	191	17	153/38

Table 1. the above table represents data size, number of classes and train samples of the respective dataset

III.MODELS

A. Text Classification Via CNN:

CNN is a type of large, feedforward neural artificial systems (where node links do not shape a cycle) whic h uses a multiple layer perceptron model built to allow negligible pre-processing and it is activated through clear cortexx in species. CNNs are commonly used in computer vision, but theyhave been applied in compliance with specific NLP requirements andthe results appear positive [2].

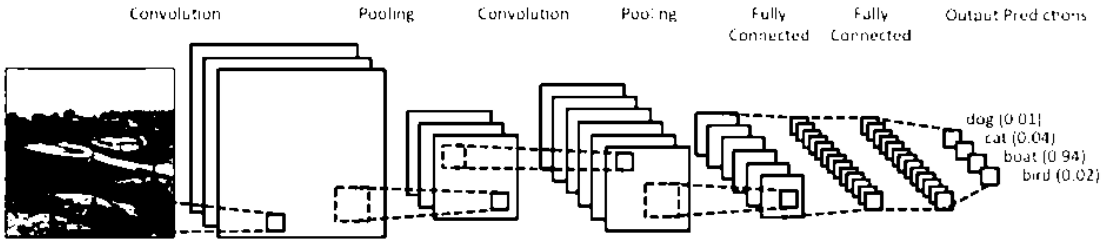

Fig2. Convolutional Neural Network[1]

We use a graphic and use CNN on text statistics. Each convolution output should fire when detecting a pa rticular pattern.

Therefore, to identify designs of different sizes (2, 3, or 5 neighboring words) by changing the scale of the kernels and concatenating their outputs. Patterns may be phrases (word sentences?) like "HELLO WORL D," "VE" and CNNs can classify them in the sentence irrespective of their location.

[1] http://www.wildml.com/wpp-content/uploadds/2015/101/Screen-Shot-2015-11-07-.png

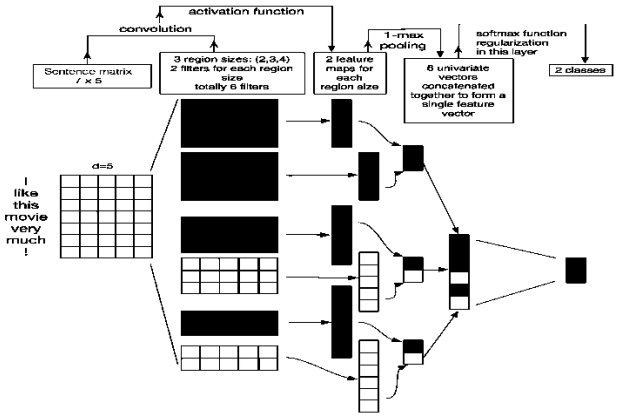

Fig3. Input as a text is passed to convolutional neural network[2]

Simplified CNN in this segment was used to create a classifier
[3]. To get rid of Hyper Text Markup Language tags and some special unsolicited characters, Beautiful S oup [4] is thus used with a view. Google Glove 6B [5] is often known as vector 100d. Glove is an unmonitored compilation of guidelines for the creation of t erm vector representations. Training is done on aggregated phrase-word co-occurrence statistics from a corpus, and the ensuing representations show case thrilling linear substructures composed space vector of the words. For an unknown word, quite simple Convolutional Architecture has been used, consisting filters of size 5 total 128 in number and 35 and 5 is the pooling max.

[2] http://www.wildml.com/wp-content/uploadds/2015/110/Screen-Shot-2015-11-06.png

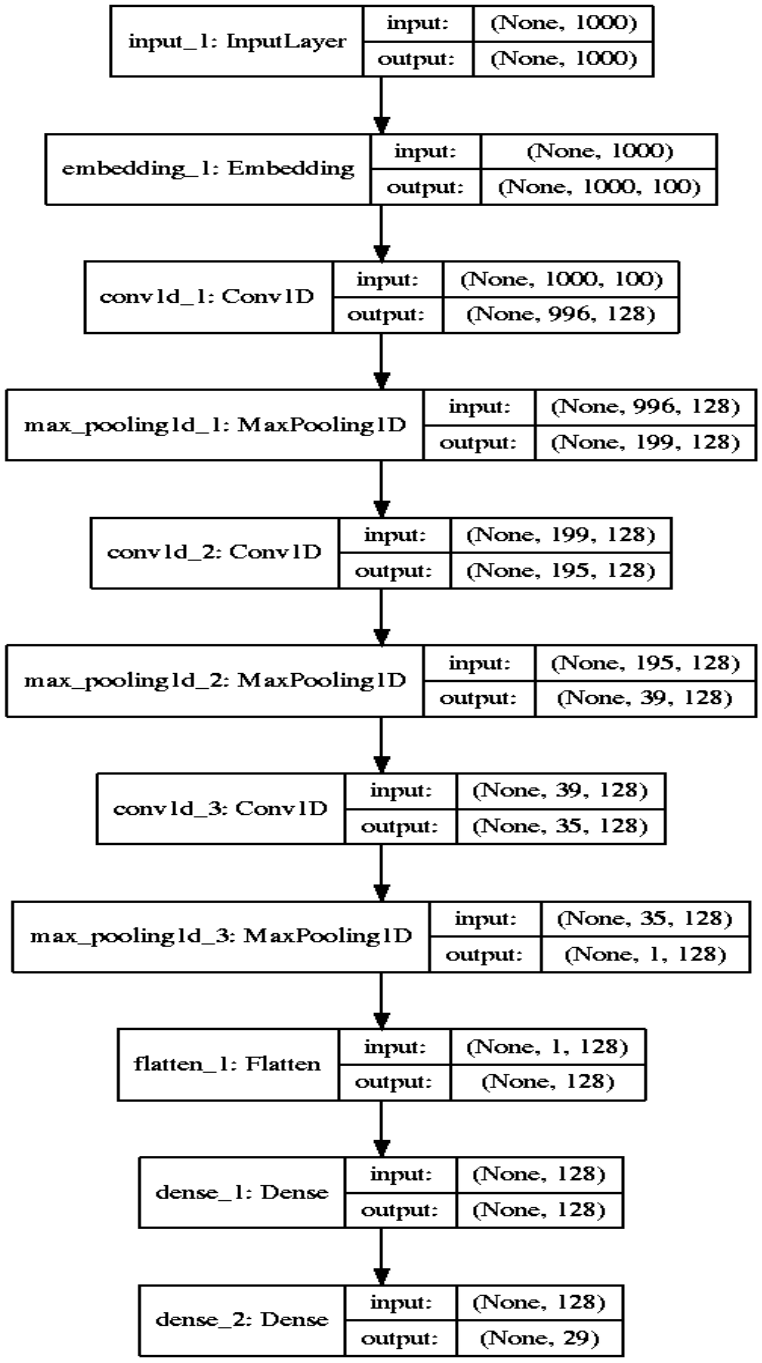

Fig 4. architecture of CNN model used for text classification

B. Text Classification Via RNN:

A recurrent neural network (RNN) is a class of synthetic neural network, networks among nodes outline into a directed graph along a series. Therefore, it permits to show non-static temporal behaviour for an order of time. It uses the knowledge from an external embedding which can remodel the precision of the RNN because it integrates new lexical and semantic data approximately the words, an facts that has been skilled and extracted on a completely vast corpus of data [6]. The Glove is used for pre-skilled embedding [7]. RNN are complicated to understand though it is moderately interesting. They encapsulate a totally lovely layout that overpowers conventional neural networks' limitations that stand up when managing collection information. RNN is a series of neuraal community blocks which might be related to every single other block and thus transferring a message to a inheritor [5, 6].

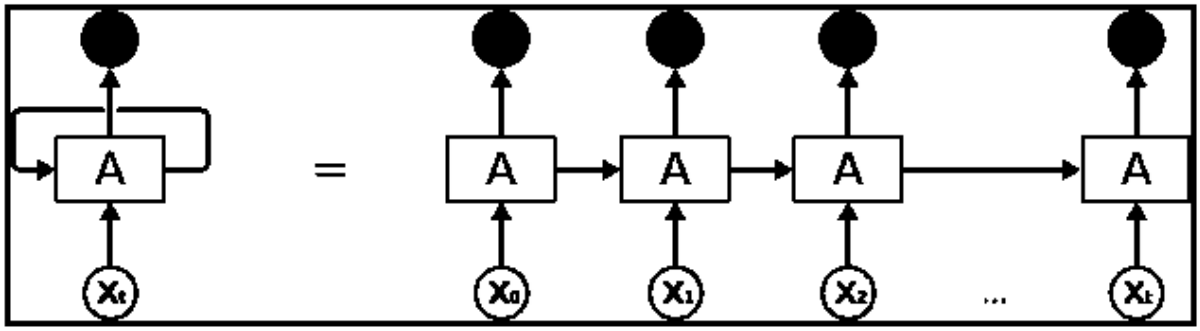

Fig 5. Recurrent Neural network Model

Beautiful Soup performs the same pre-processing as RNN [4] [6]. Therefore, text data which is sequence type is being processed by the Beautiful Soup. Kera's Tokenizer class [8] is used in order to process the Kera's on text data and num_words that is extreme amount of words reserved subsequently tokenization id done arranged the basis of the word frequency. After fitting tokenizer on the data, it is converted into strings than mapping it to sequence of numbers which represent the location of respectively word in the phrasebook. The problem is solved by using RNN and LSTM encoder which an attention based model. By means of LSTM encoder [9], all the information of text is encoded in the last output of Recurrent Neural Network before running on feedforward system for classification. This is very parallel to neural conversion machine and sequence to sequence learning. Kera's is been used which provide the wrapper called as bidirectional LSTM ,concatenating both last output of LSTM outputs.

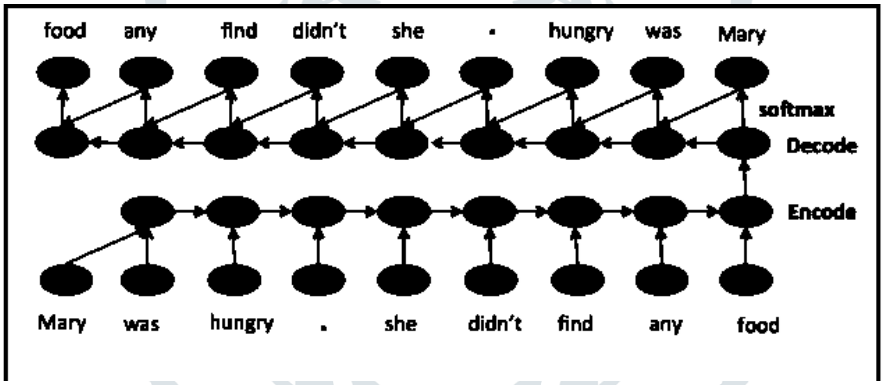

Fig 6. automatic hierarchal architectural neural network encoder.

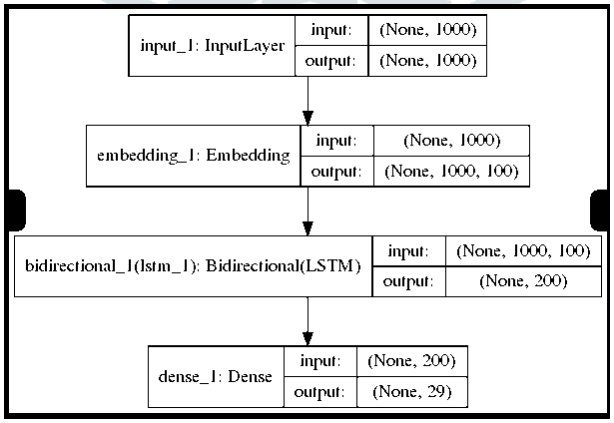

Fig7. architecture of RNN for text classification

C. Text Classification Using Hierarchical Attention Network (HAN):

Hierarchical Attention Neural network [1] is used for the document classification and processing of the document is done using Beautiful Soup [4] and for pretrained embedding we have used Glove. The 3D input is provided. Therefore, the input tensor would be [# -of evaluations each batch, # -of sentences, # -of

arguments in each sentence]. Subsequently, Kera's function Time Distributed to construct the Hierarchical input layers.

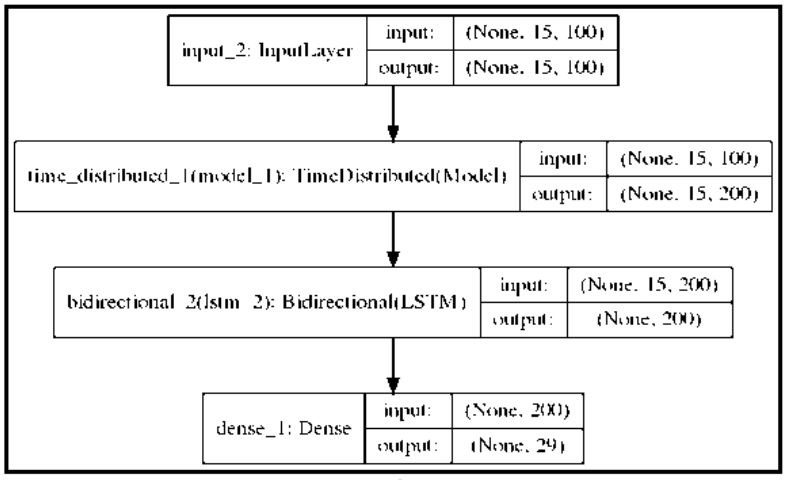

Fig 8. HAN model for text classification

III. RESULTS

The following table represents the time taken by each machine learning algorithm RNN, CNN, HAN for processing the data and also, it epitomizes the training accuracy and validation accuracy of each machine algorithm on the dataset1, dataset2, dataset3 respectively.

DATASET	ALGORITH	TIME	TRAINING ACCURACY	VALIDATION ACCURACY
DATASET 1	RNN	171 seconds	95.76	95.25
DATASET 2	RNN	1802 seconds	82.63	82.12
DATASET 3	RNN	150 seconds	89.54	92.1
DATASET 1	CNN	48 seconds	96.15	94.14
DATASET 2	CNN	360 seconds	87.88	83.76
DATASET 3	CNN	5 seconds	91.5	94.73
DATASET 1	HAN	240 seconds	94.6	94.24
DATASET 2	HAN	1400 seconds	87.32	85.54
DATASET 3	HAN	30 seconds	92.16	86.84

Table 2. the table represents the accuracy yielded by each algorithm

The resulting graph represents the accuracy against the epoch which indicates the number of passes through the entire training dataset, of dataset1, dataset2, dataset3 respectively and therefore, yielding accuracy curves that is training accuracy and validation accuracy of the respective datasets. Despite of this, loss accuracy is also calculated against the epoch of every dataset and thus, depicting the training loss and accuracy loss of respective dataset.

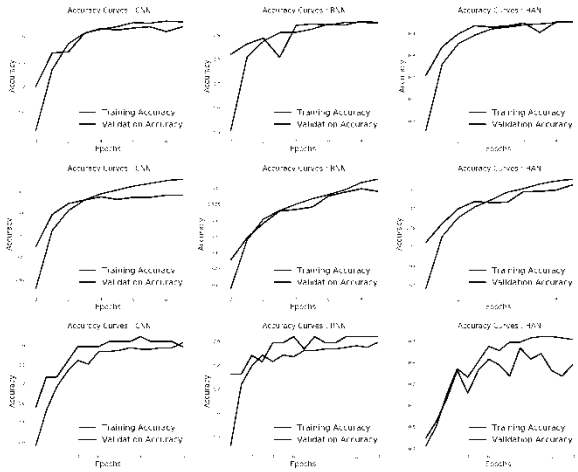

Fig 9. the picture represents accuracy curves of the RNN, CNN, HAN model for the text classification

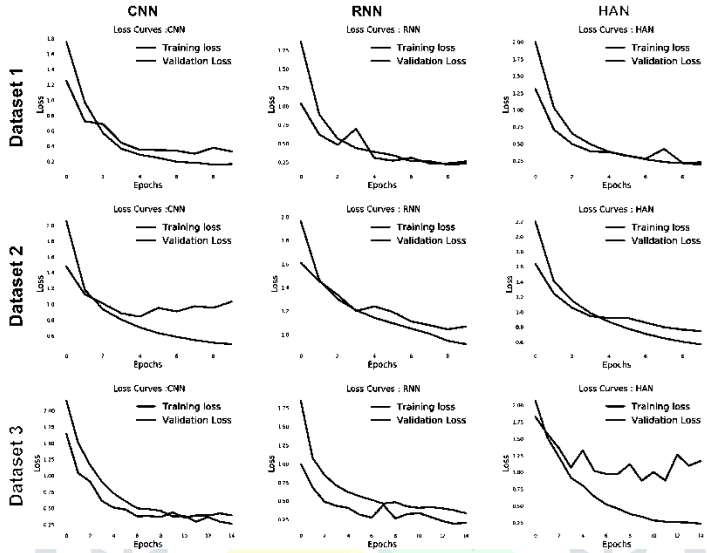

Fig10. the picture represents loss curves of the RNN, CNN, HAN model for the text classification

CONCLUSION

Based at the overhead plots, CNN has accomplished accurate validation accuracy with high reliability, also RNN & HAN have executed high accuracy but they may be no longer that constant all through all of the datasets. RNN was discovered to be the worst construction to enforce. CNN classical has outperformed the other models (RNN & HAN) in terms of training time, however HAN can carry out higher than CNN and RNN dataset is huge. For dataset 1 and dataset 2 in which the training samples are more, HAN has done the pleasant validation accuracy even as whilst the training samples are very low, then HAN has now not completed that good (dataset 3).When training samples are less (dataset 3) CNN has accomplished the fine validation accuracy.

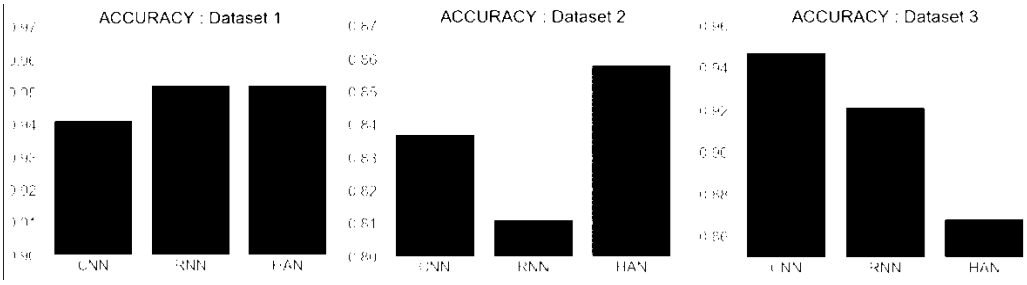

Fig 11. the picture represents the comparison of accuracy yielded by the classification models for the respective datasets.

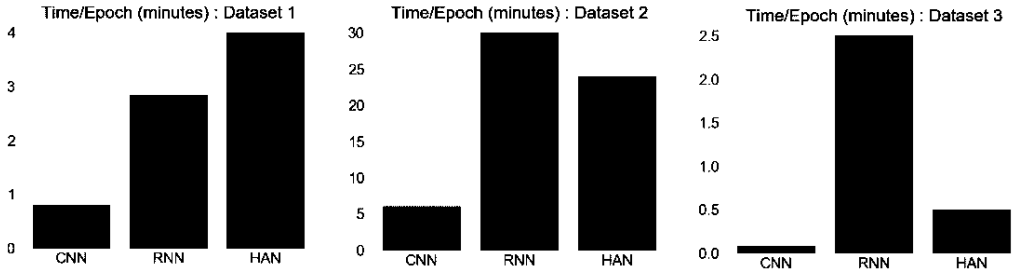

Fig 12. the picture represents the comparison of time taken by the classification models for training the respective datasets.

REFERENCES:

[1] Kim, Y. (2014). Convolutional neural networks for sentence classification. arXiv preprint arXiv:1408.5882.

[2] Kim, Yoon. "Convolutional neural networks for sentence classification." arXiv preprint arXiv:1408.5882 (2014).

[3] Peng, H., Li, J., He, Y., Liu, Y., Bao, M., Wang, L., ... & Yang, Q. (2018, April). Large-scale hierarchical text classification with recursively regularized deep graph-cnn. In Proceedings of the 2018 World Wide Web Conference (pp. 1063-1072).

[4] Gao, Q. B., & Wang, Z. Z. (2007). Center-based nearest neighbor classifier. Pattern Recognition, 40(1), 346-349.

[5] Zheng, C., He, G., & Peng, Z. (2015). A Study of Web Information Extraction Technology Based on Beautiful Soup. JCP, 10(6), 381-387.

[6] Liu, P., Qiu, X., & Huang, X. (2016). Recurrent neural network for text classification with multi-task learning. arXiv preprint arXiv:1605.05101.

[7] Huang, Q., Chen, R., Zheng, X., & Dong, Z. (2017, August). Deep sentiment representation based on CNN and LSTM. In 2017 International Conference on Green Informatics (ICGI) (pp. 30-33). IEEE.

[8] Srinivasa-Desikan, B. (2018). Natural Language Processing and Computational Linguistics: A practical guide to text analysis with Python, Gensim, spaCy, and Keras. Packt Publishing Ltd.

[9] Gao, L., Guo, Z., Zhang, H., Xu, X., & Shen, H. T. (2017). Video captioning with attention-based LSTM and semantic consistency. IEEE Transactions on Multimedia, 19(9), 2045-2055.