Deep Convolutional Arabic Sentiment Analysis With Imbalanced Data

Eslam Omara
Director of Citizens Service Department
Agriculture Directorate
Cairo Governorate
Cairo, Egypt
e_omara@hotmail.com

Mervat Mosa
Computer Science and Engineering
Faculty of Electronic Engineering
Menofia University
Menouf, Egypt
mervat_mosa@yahoo.com

Nabil Ismail
Computer Science and Engineering
Faculty of Electronic Engineering
Menofia University
Menouf, Egypt
nabil_a_ismail@yahoo.com

***Abstract*— Deep Convolutional Neural Networks (CNNs) have shown prominent performance in different NLP tasks. A basic factor in such performance is the huge amount of data used for learning. On the other hand data sets with large size, high quality and full representation needed for text analysis tasks are rarely found. In most cases existent data sets encounter class imbalance problem where data entries belong to one category is much larger than data entries represent another category. In this paper CNN performance is investigated under imbalance condition in Arabic sentiment analysis using character level representation. A data set with highly imbalanced representation of sentiment polarity classes is utilized for the investigation. Algorithm related and data related approaches are implemented to process the imbalance problem. Multiple CNNs with various configurations are tested applying cost-sensitive learning, under-sampling and over-sampling techniques.**

Keywords— Deep learning; Convolutional neural networks; Sentiment analysis; Imbalanced data.

I. Introduction

Convolutional neural networks have been applied for extracting information from raw data in natural language processing, computer vision and speech recognition. Moreover CNNs have contributed to the stat-of-the-art in numerous fields [1, 2]. CNNs are constructed of convolution, activation and max pooling layers organized in different architectures succeeded by one or multiple fully connected layers. This structure integrates both feature abstraction and classification [1]. Convolutional networks are powerful at extracting local features from time sequential data represented by one dimensional structure of temporally correlated variables [3]. In NLP, CNNs have been used for Named Entity Recognition (NER), Part–Of–Speech Tagging (POS) [4], sentiment analysis, text classification [2], sentence classification, semantic parsing, search query retrieval and sentence modeling [5].

Applications of CNNs are hindered by the availability of large annotated samples sets. A prerequisite that is costly and rare. Besides, training sets usually suffers from class imbalance state where entries numbers within two or more classes are extremely different. Imbalanced training sets can deteriorate the model performance over minority classes in both traditional classification algorithms and deep learning models [6]. In addition, if imbalanced data are combined with overlap between classes, noisy data, and insufficient training data it highly affects the classifier performance [7].

Sentiment analysis (SA) is a text processing technique that manipulates user generated opinions and classifies the text polarity as 'positive' or 'negative' [8]. SA approaches deduce sentiments expressed in text based on the language components used to compose it [9]. SA is used for analyzing views, comments and opinions to derive a conclusion about the general attitude towards a specific issue, product, service …etc. Data imbalance emerges in sentiment analysis text corpora as most opinions express a shared view of the majority who share similar common standards [10]. Sentiment classification methods belong to three types: lexicon, traditional machine learning and deep learning techniques [10, 11]. Different convolutional neural network designs have been constructed for sentiment classification based on word representation [12, 13] and character representation [2, 14].

Arabic is a language with highly complicated and wealthy morphology. A word in Arabic is a token bounded by spaces and it holds morphological information as derivation, inflection, and agglutination [15]. The language has various forms namely modern standard Arabic (MSA), dialectical Arabic and classical Arabic [11]. SA in Arabic language has been investigated using traditional machine learning algorithms [16, 17] and deep learning (DL) techniques [8]. Shallow CNNs built of one convolution layer and trained on word embedding have been applied for Arabic sentiment analysis [17, 18]. In addition, deep CNNs based on character level representation were investigated in Arabic SA [8]. Also the performance of many classifiers has been studied and assessed in a polarity detection task of a highly imbalanced Arabic short text data set employing word embedding features [11].

Character features have advantages compared to word features as low level representation help to learn any character composition [2]. Implementations of CNNs showed the capability of convolutional and pooling layers to abstract multi-level representation of input text at character level [14], but they require huge training sets especially when modeling raw low-level features as characters [2]. Also a balanced training set is recommended [2] to avoid class imbalance influence on CNNs performance [1, 10]. In a low resource language as Arabic very large balanced training sets are not available. Data sets are quite small with respect to CNNs requirements and have different distributions of opinion polarities.

The proposed paper studies the effect of imbalanced data set on deep CNNs performance based on character level representation in Arabic sentiment analysis and conducts a comparison among a baseline model (without any imbalance handling) and three different imbalance handling techniques. The first is cost-sensitive (class weight) that is used for weighting the loss function during training. Furthermore, under-sampling and over-sampling are examined. Performance measures accuracy, F-score, precision and recall are evaluated for the implemented architectures. The recommended measure used with imbalanced data is F-score [19, 11]. F-value evaluates the classifier efficiency in case of rare classes' existence [20].

The paper is organized as follows: section II highlights data imbalance problem. Related work is presented in section III. In section IV CNN design is discussed. Data preparation and experiments results are reported in section V. Conclusion and future extension are investigated in section 0.

II. IMBALANCED CLASSIFICATION

Imbalanced classification refers to a classification task in imbalanced domain where instances of the data set are not equally distributed among categories. The problem distribution is skewed resulting in minority and majority classes. Imbalanced distribution is described by the imbalance ratio (IR) defined as the majority class entries divided by the minority class entries [7]. The case of imbalanced data set may emerge due to concept occurrence rarity or existence of restrictions during the collection of a particular class data [7, 21]. Class imbalance influences training convergence and test generalization [1].

In imbalanced classification, algorithm performance is deviated towards the majority class where the accuracy is computed according to correctly classified examples. For skewed distributions results are misleading as minority instances are considered rare occurrences, ignored, or supposed to be noise or outliers. The deviation towards the majority class occurs in the learning phase that achieves high accuracy and generalization at the expense of the minority class that is not modeled efficiently [7, 22]. Common solutions introduced to manipulate imbalanced classification are divided into data related and algorithm related methodologies.

Algorithmic approaches fine tune the classification algorithm to enhance the learning process which boosts the algorithm sensitivity to the minor class [7, 21, 22]. Different algorithmic techniques have been investigated as thresholding approach that is carried out in the test phase and adjusts the network output. The output category probability is recalculated using prior probabilities computed based on class occurrence in the imbalanced data set. Cost-sensitive learning is a methodology which defines different costs for misclassified examples from various classes. During learning higher cost classes have more effect on weights updates. Another method is One-class classification that learns to identify the interest class rather than distinguishing between classes. For CNNs two-phases training has also been investigated where a training phase on a balanced data set is conducted followed by a fine tuning phase of the output layers [1, 23].

Data level solutions basically depend on sampling techniques and commonly used as their application is independent of the classifier algorithm [23, 10]. Three sampling schemes are widely utilized (Under-sampling, Over-sampling, Hybrid techniques) to cope with standard classifiers assumption that is the domain data set is balanced [7, 21]. Random under-sampling is the simplest approach which deletes elements from the majority class to rebalance the class distribution. Under-sampling may ignore considerable instances in the training data. Random Over-sampling repeats occurrences of minority cases. But over-sampling may increase the probability of overfitting as it supports the minority class and ignores the actual representation in the problem. Hybrid techniques combine both over and under sampling approaches [7, 21].

III. RELATED WORK

The effect of class imbalance on CNNs performance has been investigated in image processing [1, 24, 6, 23]. Common data approaches as over-sampling, under-sampling were implemented and compared. Moreover, algorithm techniques as cost sensitive, new loss functions, two-phase training and thresholding have been applied. It was highlighted that imbalanced sets degrades CNNs performance and over-sampling is a prominent technique. Complete over-sampling could handle the imbalance problem while implementing under-sampling depends on the imbalance ratio. In addition it was noted that over-sampling does not cause CNNs overfitting [1].

Imbalance problem has been studied within text data applying different classical algorithms as Naive Bayes (NB), Logistic Regression (LR) and Support Vector Machine (SVM) [19, 10, 11, 27]. Deep learning architectures have also been experimented as Bidirectional Long Short-term Memory (LSTM), Recurrent Neural Network (RNN) and Convolutional Neural Network (CNN), [19, 10, 25]. New loss function, pruning convolution filters and cost-sensitive learning approaches have been proposed and tested [23, 26, 25]. Data level methodologies have been verified as under-sampling, over-sampling and synthetic minority over-sampling which adds artificial samples to the minority class depending on distance similarities among minority class entries in the feature space [11]. Data augmentation is exploited to create synthetic examples used to over-sample minority category. Different preprocessing techniques are conducted to generate new instances as removing duplicated words from text entries, randomly deleting a percent of the original words or substituting a number of the original text words by their synonyms and antonyms [19, 10].

IV. CONVOLUTIONAL NEURAL NETWORK

A. Network design

Character-level convolutional network applied for CNN performance evaluation in Arabic sentiment classification follows the architecture proposed in [2] and investigated in Arabic sentiment analysis in [8]. The network is built of two temporal modules one-dimensional convolution and one-dimensional max-pooling. The deep structure combines six convolution layers, three max pooling layers, two dropout modules and three fully connected layers. To perform the assessment and the comparison taking into account the network size, three CNNs have been built and tested namely networks (A, B, C) with (128, 256, 1024) feature maps respectively. The design of network (B) is shown in Fig.1

B. Network settings

An encoded character sequence is fed to the network input layer. To quantize characters (1-of-m) encoding is used where m is the character vocabulary size. A list of (36) characters with the highest occurrence in the corpus are selected to form the alphabet. Out of the list characters are coded as zero vectors. The vocabulary list is:

(ء ا آ أ إ ب ة ت ث ج ح خ د ذ ر ز س ش ص ض ط ظ ع غ ف
ق ك ل م ن ه و ؤ ى ي ئ)

Text instances are converted to vectors of length (l = 1014) of encoded characters. Large instances are pruned and small instances are padded with zero [2, 8]. Implementation is conducted on Python, Keras and Tensorflow. CUDA9 and CUDNN7 are used for CNN layers. The network training is carried out utilizing NVIDIA GEFORCE GTX 1070 GPU. The settings applied are the same in [2, 8]. Class weight is computed by Scikit-learn (Sklearn) library with balance option [28]. Random over-sampling and random under-sampling are implemented by imbalanced-learn (Imblearn) library [29]. Stochastic Gradient Descent (SGD) with momentum 0.9 is used for training. A learning rate of 0.01 is defined and halved with triple epochs. Every epoch processes 5000 steps of stochastically selected instances. A batch of 128 instances is fed to the network trained for 50 epochs.

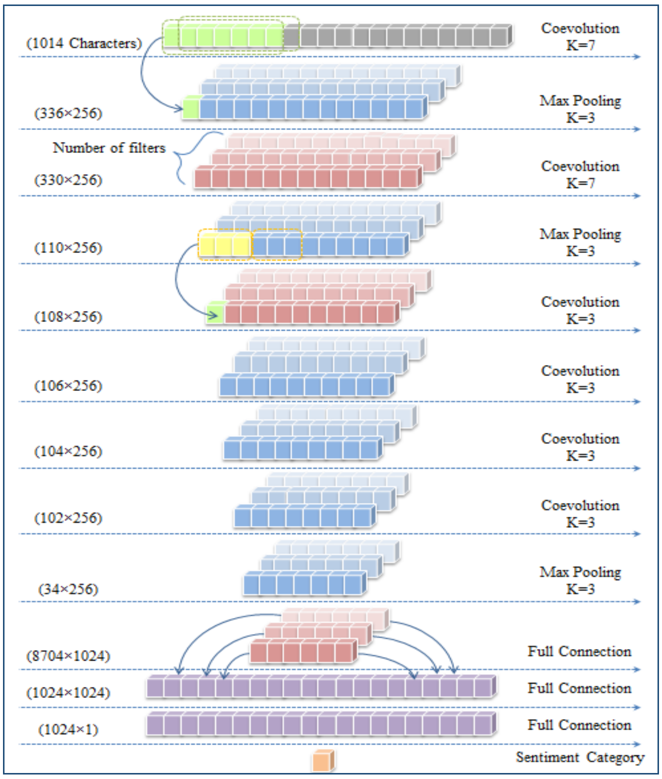

FIG. 1. CONVOLUTIONAL NEURAL NETWORK DESIGEN

V. EXPERIMENTS AND RESULTS

A. Data preparation

In order to evaluate CNNs performance applying the different balancing techniques, the unbalanced data set constructed from Arabic sentiment analysis pre-annotated data sets proposed in [8] is used. The data set is a combination of thirteen sets from various domains (Book Reviews, Tweets, Product Reviews, Hotel Reviews, Movie Reviews, Restaurant Reviews) represented in Modern Standard (MSA) and Dialectal Arabic.

TABLE I. MANUPILATING IMBALANCED DATA SETS REVIEW

Ref	Field	Network	Techniques	Notes
[1]	Image Classification	Convolutional neural network (CNN)	Over-sampling Under-sampling Two-phase training Thresholding	Over-sampling is prominent No overfitting with over-sampling
[24]	Image Classification	Convolutional neural network (CNN)	Cost-sensitive learning	Class-dependent costs rely on training data statistics
[6]	Image Classification	Convolutional neural network (CNN)	Deep Over-sampling (DOS)	Re-sample training data in the linear subspace of deep features
[23]	Image Document Classification	Deep neural network (DNN)	New loss functions	Proposed functions precede mean squared error (MSE) on high imbalanced data
[26]	Text Classification	Convolutional neural network (CNN)	Pruning convolution filters	Optimized CNN filters efficiently identify minority classes in extremely imbalanced data
[19]	Comment Toxicity Classification	Ensemble (Convolutional neural network (CNN), Bidirectional long short-term memory (LSTM), Bidirectional gated Recurrent units (GRU))	Data augmentation	Ensemble model performance exceeds individual applied networks on imbalanced text corpus
[10]	Sentiment Classification	Convolutional neural network (CNN) Recurrent neural network (RNN) Support vector machine (SVM) Naive Bayes (NB) Logistic regression (LR)	Data augmentation (Generating samples from text entries based on discourse markers)	Integrates data augmentation with over-sampling
[11]	Sentiment Analysis	Traditional classifiers Ensemble	SMOTE (Synthetic Minority Over-sampling Technique)	Stacking ensemble combined with SMOTE on word embedding features showed high performance
[27]	Sentiment Classification	K-nearest neighbors (KNN) Naive Bayes (NB) Random Forest (RF) Support Vector Machine (SVM)	Under-sampling Over-sampling SMOTE	Used binarization, dimensionality reduction and sampling techniques
[25]	Student level prediction from forum texts	Convolutional neural network (CNN)	Adapted loss function Cost-sensitive learning	Adopted loss function CNN empowers predictions of minority class instances

The raw data set contains (92492) entries and the preprocessed set includes (92123) entries where different text preprocessing steps were executed [8].

B. Results analysis

The base networks without imbalance handling and the networks augmented with imbalance manipulation techniques are evaluated upon both raw and preprocessed sets. The data sets have highly imbalanced representation of the polarity classes (positive, negative). Almost 77 percent of the text instances belong to the positive class as shown in table II. Three techniques are carried out to handle the imbalance problem where algorithm related and data related methods are investigated. For cost-sensitive learning classes weights are pre-defining to be used for weighting the loss function in the learning phase. This technique enables the network to assign more weights to entries belong to an improperly represented category [30]. Besides, majority class under-sampling and minority class over-sampling are tested and compared.

TABLE II. PREPROCESSED DATA SET FOR ARABIC SENTIMENT ANALYSIS

Category	Total Entries	Percent
Positive	71112	77.2
Negative	21011	22.8

F-score, precision, recall and accuracy are evaluated for the three CNNs architectures with different number of feature maps. For imbalanced data sets F-score is a more informational measure calculated as the weighted average of both precision and recall [19, 11]. TABLE III shows the F-score for the applied networks. The effects of imbalance handling techniques on negative class and positive class F-values are shown in Fig.2 and Fig.3. Calculated performance measures show that over-sampling is an effective approach within CNN classifiers designed for Arabic sentiment analysis depending on character representation.

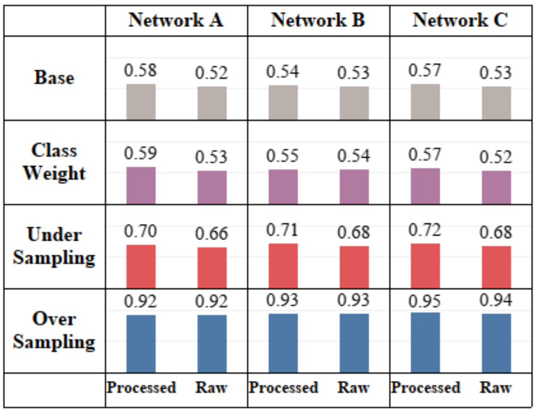

FIG. 2. F-SCORE FOR NEGATIVE CLASS

TABLE III. NETWORKS F-SCORE

Model	Data set	Base	Class Weight	Under Sampling	Over Sampling
Model A	Processed	0.81	0.82	0.69	0.92
(128 Filters)	Raw	0.79	0.79	0.66	0.92
Model B	Processed	0.80	0.80	0.71	0.93
(256 Filters)	Raw	0.80	0.80	0.68	0.92
Model C	Processed	0.81	0.81	0.71	0.94
(1024 Filters)	Raw	0.80	0.80	0.68	0.94

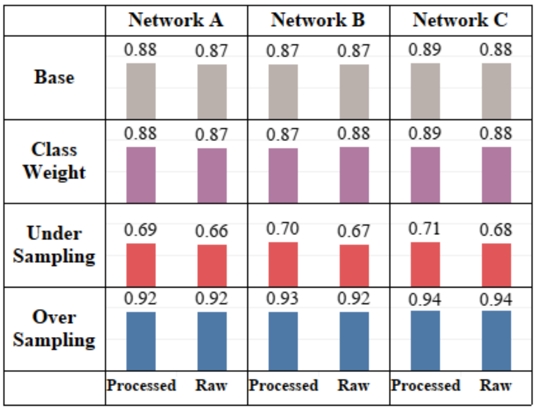

FIG. 3. F-SCORE FOR POSITIVE CLASS

The application of CNN for Arabic SA employing character features with imbalance handling, registered the accuracy values shown in table IV for the three networks and the three imbalance handling methodologies. Results shows that over-sampling approach reported higher accuracy values. Computed performance measures for the two sets (preprocessed, raw) illustrate that large deep CNNs are able to abstract features from character level encoded text.

TABLE IV. NETWORKS ACCURACY

	Data set	Base	Class Weight	Under Sampling	Over Sampling
Network A	Processed	81.73	82.02	69.36	92.08
(128 Filters)	Raw	78.92	79.86	65.98	92.13
Network B	Processed	80.37	80.19	70.57	92.90
(256 Filters)	Raw	80.11	80.52	67.78	92.50
Network C	Processed	82.18	82.12	71.35	94.33
(1024 Filters)	Raw	80.64	82.06	71.37	94.41

Filters learned by the first convolutional layer of Network C using raw set are displayed in Fig.4. The learned filters are connections weights. The first five filters with kernel equals seven of the (1024) feature maps are plotted and each input channel of the 37 (36 characters and space) is drawn as a column. The dark blue squares define small weights and the dark yellow squares declare large weights that detect the feature associated with them.

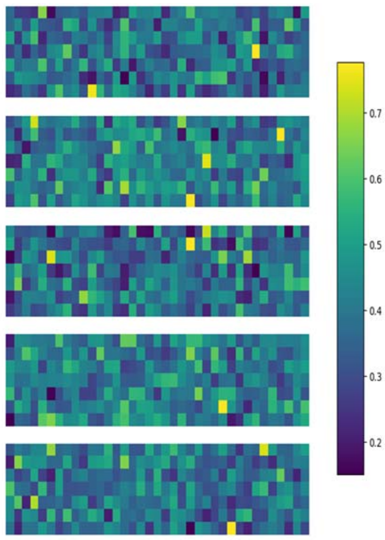

FIG. 4. THE FIRST FIVE LEARNED FILTERS

Performance measures Recall, Precision and F-score evaluated for the implemented architectures using both raw and processed data sets are shown in the next tables. Equations 1, 2, 3 and 4 define calculation formulas [11, 20].

$$\text{Precision} = (TP/ (TP+ FP)) \quad (1)$$
$$\text{Recall} = (TP/ (TP+ FN)) \quad (2)$$
$$\text{Accuracy} = ((TP+TN) / (TP+TN+ FP+FN)) \quad (3)$$
$$\text{F-score} = ((\text{Precision} \times \text{Recall}) / (\text{Precision} + \text{Recall})) \times 2 \quad (4)$$

Where:
TP is the number of true positive instances.
TN is the number of true negative instances.
FP is the number of false positive instances.
FN is the number of false negative instances.

Further investigation of performance measures shows that under-sampling affects negatively the overall Precision, Recall and F-score of deep CNNs. In addition it shows the least values compared to other balancing approaches.

TABLE V. MEASURES OF NETWORK A USING PROCESSED DATA SET

		Precision	Recall	F-score	Support
Base	Negative	0.61	0.55	0.58	4202
	Positive	0.87	0.90	0.88	14223
	Average / Total	0.81	0.82	0.81	18425
Class Weight	Negative	0.62	0.57	0.59	4202
	Positive	0.87	0.90	0.88	14223
	Average / Total	0.82	0.82	0.82	18425
Under Sampling	Negative	0.69	0.70	0.70	4203
	Positive	0.70	0.69	0.69	4202
	Average / Total	0.69	0.69	0.69	8405
Over Sampling	Negative	0.89	0.97	0.92	14223
	Positive	0.96	0.88	0.92	14222
	Average / Total	0.92	0.92	0.92	28445

TABLE VI. MEASURES OF NETWORK A USING RAW DATA SET

		Precision	Recall	F-score	Support
Base	Negative	0.54	0.49	0.52	4219
	Positive	0.85	0.88	0.87	14280
	Average / Total	0.78	0.79	0.79	18499
Class Weight	Negative	0.57	0.51	0.53	4219
	Positive	0.86	0.89	0.87	14280
	Average / Total	0.79	0.80	0.79	18499
Under Sampling	Negative	0.66	0.65	0.66	4219
	Positive	0.66	0.67	0.66	4219
	Average / Total	0.66	0.66	0.66	8438
Over Sampling	Negative	0.88	0.97	0.92	14280
	Positive	0.97	0.87	0.92	14280
	Average / Total	0.93	0.92	0.92	28560

TABLE VII. MEASURES OF NETWORK B USING PROCESSED DATA SET

		Precision	Recall	F-score	Support
Base	Negative	0.58	0.51	0.54	4202
	Positive	0.86	0.89	0.87	14223
	Average / Total	0.80	0.80	0.80	18425
Class Weight	Negative	0.57	0.53	0.55	4202
	Positive	0.86	0.88	0.87	14223
	Average / Total	0.80	0.80	0.80	18425
Under Sampling	Negative	0.70	0.71	0.71	4203
	Positive	0.71	0.70	0.70	4202
	Average / Total	0.71	0.71	0.71	8405
Over Sampling	Negative	0.89	0.97	0.93	14223
	Positive	0.97	0.88	0.93	14222
	Average / Total	0.93	0.93	0.93	28445

TABLE VIII. MEASURES OF NETWORK B USING RAW DATA SET

		Precision	Recall	F-score	Support
Base	Negative	0.57	0.49	0.53	4219
	Positive	0.86	0.89	0.87	14280
	Average / Total	0.79	0.80	0.80	18499
Class Weight	Negative	0.58	0.51	0.54	4219
	Positive	0.86	0.89	0.88	14280
	Average / Total	0.80	0.81	0.80	18499
Under Sampling	Negative	0.67	0.69	0.68	4219
	Positive	0.68	0.67	0.67	4219
	Average / Total	0.68	0.68	0.68	8438
Over Sampling	Negative	0.89	0.97	0.93	14280
	Positive	0.97	0.88	0.92	14280
	Average / Total	0.93	0.93	0.92	28560

TABLE IX. MEASURES OF NETWORK C USING PROCESSED DATA SET

		Precision	Recall	F-score	Support
Base	Negative	0.64	0.51	0.57	4202
	Positive	0.86	0.91	0.89	14223
	Average / Total	0.81	0.82	0.81	18425
Class Weight	Negative	0.63	0.52	0.57	4202
	Positive	0.86	0.91	0.89	14223
	Average / Total	0.81	0.82	0.81	18425
Under Sampling	Negative	0.71	0.72	0.72	4203
	Positive	0.72	0.70	0.71	4202
	Average / Total	0.71	0.71	0.71	8405
Over Sampling	Negative	0.91	0.98	0.95	14223
	Positive	0.98	0.91	0.94	14222
	Average / Total	0.95	0.94	0.94	28445

TABLE X. MEASURES OF NETWORK C USING RAW DATA SET

		Precision	Recall	F-score	Support
Base	Negative	0.59	0.48	0.53	4219
	Positive	0.85	0.90	0.88	14280
	Average / Total	0.80	0.81	0.80	18499
Class Weight	Negative	0.60	0.46	0.52	4219
	Positive	0.85	0.91	0.88	14280
	Average / Total	0.79	0.81	0.80	18499
Under Sampling	Negative	0.68	0.68	0.68	4219
	Positive	0.68	0.68	0.68	4219
	Average / Total	0.68	0.68	0.68	8438
Over Sampling	Negative	0.91	0.98	0.94	14280
	Positive	0.98	0.90	0.94	14280
	Average / Total	0.94	0.94	0.94	28560

Over-sampling, under-sampling and class weight promotes the negative minority class scores. On the other hand, under-sampling downgrades the positive majority class measures. With respect to the base model, class weight slightly enhances performance while over-sampling largely fosters it. Data balancing techniques reshape the class distribution and hence the model realizes a comparable performance on both classes which interprets the similarity of calculated values for positive and negative categories applying over and under sampling.

VI. CONCLUSION

Deep CNNs are powerful features extractors that mainly depend on large data sets. Such a data resource is rarely found especially in low resource languages. In most cases the availability is combined with class imbalance problem. The implementation of imbalance handling techniques shows that random over-sampling enhances CNNs performance compared to other applied approaches. The network (C) with large number of feature maps reported accuracy (94.41) and F-score (0.94) higher than networks (A, B) trained on the same imbalanced data set and supported with identical manipulation technique. Besides, the registered measures for each network are very close on both processed and raw sets (for example 0.93 and 0.92 F-score for network B). That clarifies CNNs capability to abstract features from raw text data. Other balancing approaches can be more investigated applying deep convolutional networks and employing word and character representation. Furthermore the effect of imbalance handling methods may be studied and compared using various deep learning architectures fed with different text representation methodologies.

References

[1] M. Buda, A. Maki, and M.A. Mazurowski, “A systematic study of the class imbalance problem in convolutional neural networks,” Neural Networks, vol. 106, pp. 249-259, 2018.

[2] X. Zhang, J. Zhao and Y. LeCun, “Character-level convolutional networks for text classification,” In Advances in neural information processing systems, pp. 649-657, 2015.

[3] Y. LeCun, and Y. Bengio, “Convolutional networks for images, speech, and time series,” The handbook of brain theory and neural networks, vol. 3361, no. 10, 1995.

[4] Y. Chen, “Convolutional neural network for sentence classification,” Master's thesis, University of Waterloo, 2015.

[5] Y. Kim, “Convolutional neural networks for sentence classification,” In Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP), pp.1746–1751, October 2014.

[6] S. Ando, and C.Y. Huang, “Deep over-sampling framework for classifying imbalanced data,” In Joint European Conference on Machine Learning and Knowledge Discovery in Databases, Springer, pp. 770-785, September 2017.

[7] A. Fernández, S.D Río, N.V. Chawla, and F. Herrera, “An insight into imbalanced Big Data classification: outcomes and challenges,” Springer, vol. 3, pp. 105–120, 2017.

[8] E. Omara, M. Mousa and N. Ismail, “Deep Convolutional Network for Arabic Sentiment Analysis,” International Japan-Africa Conference on Electronics, Communications and Computations (JAC-ECC), IEEE, pp. 155-159, December 2018.

[9] M. Al-Ayyoub, A.A. Khamaiseh, Y. Jararweh and M.N. Al-Kabi, “A comprehensive survey of arabic sentiment analysis,” Information Processing & Management, vol. 56, no. 2, pp .320-342, 2019.

[10] T. Zhang, X. Wu, M. Lin, J. Han, and S. Hu, “Imbalanced Sentiment Classification Enhanced with Discourse Marker,” arXiv:1903.11919, 2019.

[11] S. Al-Azani and E.S.M. El-Alfy, “Using Word Embedding and Ensemble Learning for Highly Imbalanced Data Sentiment Analysis in Short Arabic Text,” Procedia Computer Science, vol. 109C, pp. 359-366, 2017.

[12] N. Kalchbrenner, E. Grefenstette and P. Blunsom, ”A convolutional neural network for modelling sentences,” arXiv:1404.2188, 2014.

[13] B. Shin, T. Lee and J.D. Choi, “Lexicon integrated cnn models with attention for sentiment analysis,” arXiv:1610.06272, 2016.

[14] A. Conneau, H. Schwenk, Y.L. Cun and L. Barrault, “Very Deep Convolutional Networks for Text Classification,” In Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics, vol. 1, pp 1107–1116, 2017.

[15] N. Boudad, R. Faizi, R.O.H Thami and R. Chiheb, “Sentiment analysis in Arabic: A review of the literature,” Ain Shams Engineering Journal, 2017.

[16] A.A. Sallab, H. Hajj, G. Badaro, R. Baly, W.E. Hajj, and K.B. Shaban, “Deep learning models for sentiment analysis in Arabic,” In Proceedings of the Second Workshop on Arabic Natural Language Processing, pp. 9-17, 2015.

[17] A. Dahou, S. Xiong, J. Zhou, M.H. Haddoud and P. Duan, “Word embeddings and convolutional neural network for arabic sentiment classification,” In Proceedings of COLING 2016 the 26th International Conference on Computational Linguistics, pp. 2418-2427 , 2016.

[18] A.M. Alayba, V. Palade, M. England and R. Iqbal, “Improving Sentiment Analysis in Arabic Using Word Representation,” arXiv preprint arXiv:1803.00124, 2018.

[19] M. Ibrahim, M. Torki and N. El-Makky, “Imbalanced Toxic Comments Classification Using Data Augmentation and Deep Learning,” 17th IEEE International Conference on Machine Learning and Applications (ICMLA), IEEE, pp. 875-878, December 2018.

[20] N.V. Chawla, “Data mining for imbalanced data sets: An overview,” In Data Mining and Knowledge Discovery Handbook, Springer, pp. 875-886, 2009.

[21] R. Longadge, S. S. Dongre, L. Malik, “Class Imbalance Problem in Data Mining: Review,” International Journal of Computer Science and Network (IJCSN), vol. 2, no. 1, February 2013.

[22] A. Ali, S. M. Shamsuddin and A. L. Ralescu, “Classification with class imbalance problem: A Review,” Int. J. Advance Soft Compu. Appl, vol. 7, no. 3, pp. 176-204, November 2015.

[23] S. Wang, W. Liu, J. Wu, L. Cao, Q. Meng and P.J Kennedy, “Training deep neural networks on imbalanced data sets,” International Joint Conference On Neural Networks (IJCNN), IEEE, pp. 4368-4374, July 2016.

[24] S.H. Khan, M. Hayat, M. Bennamoun, F.A. Sohel and R. Togneri, “Cost-sensitive learning of deep feature representations from imbalanced data,” IEEE Transactions on Neural Networks And Learning Systems, vol. 29, no. 8, pp. 3573-3587, 2018.

[25] P.H.G. Nguyen and C.T.N. Vo, “A CNN Model with Data Imbalance Handling for Course-Level Student Prediction Based on Forum Texts,” International Conference on Computational Collective Intelligence, Springer, pp. 479-490, September 2018.

[26] H.J. Yoon, S. Robinson, J.B. Christian, J.X. Qiu and G.D. Tourassi, “Filter pruning of Convolutional Neural Networks for text classification: A case study of cancer pathology report comprehension,” IEEE EMBS International Conference on Biomedical & Health Informatics (BHI), IEEE, pp. 345-348, March 2018.

[27] B. Krawczyk, B.T. McInnes and A. Cano, “Sentiment classification from multi-class imbalanced twitter data using binarization,” International Conference on Hybrid Artificial Intelligence Systems, Springer, pp. 26-37, June 2017.

[28] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg and J. Vanderplas, “Scikit-learn: Machine learning in Python,” Journal of machine learning research, pp. 2825-2830, October 2011. (https://scikit-learn.org/stable/)

[29] G. Lemaître, F. Nogueira and C.K. Aridas, “Imbalanced-learn: A python toolbox to tackle the curse of imbalanced data sets in machine learning,” Journal of Machine Learning Research, vol. 18, no. 1, pp. 559-563, 2017. (https://imbalanced-learn.readthedocs.io/en/stable/)

[30] F. Chollet, “Keras documentation," keras. io, 2015. (https://keras.io)