J. Pokojski et al. (Eds.)

doi:10.3233/ATDE200094

Impact Assessment of Food Safety News Using Stacking Ensemble Learning

Bo SONG[a], Kefan Shang[b] Junliang HE[a,1], Wei YAN[a] and Tianjiao ZHANG[c]
[a] *China Institute of FTZ Supply Chain, Shanghai Maritime University, Shanghai 201306, P. R. China*
[b] *Logistics Engineering Colleage, Shanghai Maritime University, Shanghai 201306, P. R. China*
[c] *Colleage of Information Technology, Shanghai Ocean University, Shanghai 201306, P. R. China*

Abstract. Food safety has always been the focus of public concern. Assessment of the impact of food safety news constituents an important job of the government departments. In this paper we present a method using stacking ensemble learning to assess the impact level of global food safety news. The news used for training the assessment model is collected from the Chinese customs. Each of the news articles is annotated with a label ranging from low impact, medium impact and high impact by the customs officials. For base learners in the ensemble learning model, we use Naive Bayesian, Support Vector Machine, XGBoost, FastText, Convolutional Neural Network, LSTM and BERT. A Naive Bayesian-based meta learner is used to integrate the assessment results of the base learners. The proposed method features end to end prediction of news impact using the original news text as input, and it advances the transdisciplinary development of artificial intelligence and risk assessment by improving the accuracy of impact assessment of food safety news compared with traditionally used methods.

Keywords. Stacking ensemble learning, impact assessment, food safety news

Introduction

Food safety concerns the health of people and therefore keeps receiving a lot of attentions from both the public and the government. To monitor the news reporting food safety events and assess the impact of the news constitute one of the important work of the government. With the fast development of artificial intelligence (AI) , the automatic assessment of food safety news using AI techniques such as natural language processing and machine learning becomes realistic and is desired by regulators of the government to improve supervision efficiency. Since food safety news is usually related with adverse events such as food disqualification or detection of harmful substances in food, the assessment of food safety news is similar with assessing the food risks described in the news. In this sense, automatic food safety news assessment becomes a transdisciplinary engineering problem incorporating AI and risk assessment. For risk assessment, it usually comprises recognizing the consequences of adverse events, assessing the probability and severity of each consequence, and synthesizing

[1] Corresponding Author, Mail: jlhe@shmtu.edu.cn.

the assessed consequences to get the final result [1]. For example, Fu et al. [2] modeled a risk scenario as an event tree, and then carried out dependency analysis of the associated intermediate events to quantify the probability of event occurrence. Using a quantitative risk assessment method called the Fine Kinney method, Kokangul et al. [3] computed risk as the likelihood of hazardous event multiplies the exposure factor multiplies the possible consequence. This kind of risk assessment makes the assessment process highly depended on expert knowledge and formal representation of risk indexes, therefore lacking the ability to react swiftly to the varying risk situations in the big data era. Recently, there is a trend of assessing risks with textual materials as input [4,5]. For example, Su and Chen [4] assessed the risk in global suppler selection using information mined from Twitter. Duy et al. [5] predicted mental health risk through automated analysis of case notes in electronic health records. For food safety news, it consists of unstructured text with various themes which is difficult to be represented formally. If a model were to assess the impact of food safety news using explicit indexes such as food type, harmful ingredient and region of influence, then a series of dictionaries and corresponding information extractors should be developed, and the assessment result would be affected by the incompleteness of each dictionary and imprecision of each information extractor. In this paper, we propose to assess news impact in an end to end fashion, which is to use an ensemble machine learning model to learn the mapping of news articles to impact grades directly. Such a fashion is similar to the real news assessment process in the Chinese customs, where customs officials grade the impact of a news article immediately after reading it based on their experience. To learn the complex relationship between news articles and impact grades, we first train seven different base classifiers to classify news articles into three categories: low impact, medium impact and high impact. Then a meta classifier is trained with the output of base classifiers to generate the final prediction of impact level. The base classifiers are Naive Bayesian, Support Vector Machine (SVM), XGBoost, FastText, Convolutional Neural Network (CNN), LSTM and BERT. Naive Bayesian is used as the meta classifier.

1. News impact assessment as a text classification task

As a daily routine, the Chinese customs collects food safety news around the world and rated it with three categories: low impact news, medium impact news and high impact news. For low impact news, it is relevant to food import/export but poses little risk for China. News with medium impact may involve announcement of disqualified food, food recall or change of import/export regulations. High impact news is usually related with detection of harmful substances in popularly used imported food, or rejection/detention of Chinese food by other countries. However, the categorization of food safety news cannot be simply done by following the above rules of thumb. This is because the types of food safety news are not restricted to the abovementioned ones and for each type of news, the impact level of the news may vary according to the involved food types, affected countries/areas, consequence of event, economic/political issues and so on. Considering these complex features of food safety news, we believe that a more efficient way for news assessment would be to directly classify a news article in to low, medium and high impact classes using word-based features. Using word-based features enables us to draw upon the latest techniques in text classification and at the same time avoid extraction of complex features which leads to poor reliability and scalability.

Text classification is an intensively studied research field. According to a recent survey, frequently used text classification techniques include Naive Bayesian, SVM, XGBoost, FastText, CNN and RNN (Recurrent Neural Network) [6]. Text classifiers can benefit from proper quantification of text strings. For example, using TF-IDF scores as the weights of words can help a classifier to discriminate between informative words and uninformative words. Word embedding is a technique of representing words with short and dense vectors (compared with one-hot representation of words which is long and sparse) to reflect the semantic meaning of words and improve computational efficiency [7]. Word2Vec, a widely used word embedding method, maps high dimensional one-hot representation of words to a low dimensional vector space by training a neural network [8]. In this paper, we use TF-IDF as the word weights for text classifiers using bag-of-words as input, and use Word2Vec to generate word vectors for neural network-based classifiers.

2. Ensemble machine learning

Although various machine learning methods exist for text classification, it is hard to determine in advance which method performs the best for the given situation. To solve this problem, researchers proposed ensemble learning, a framework of integrating multiple machine learning models. In the context of this paper, ensemble learning is to integrate multiple text classifiers. The text classifiers to be integrated are called the base classifiers, which can be of different types, or of the same type but trained differently. By summarizing the results of the base classifiers, the overall classification result can be obtained, and the accuracy and generalization ability of the ensemble classifier can be improved [9]. Figure 1 shows the framework of the ensemble classifier proposed in this paper.

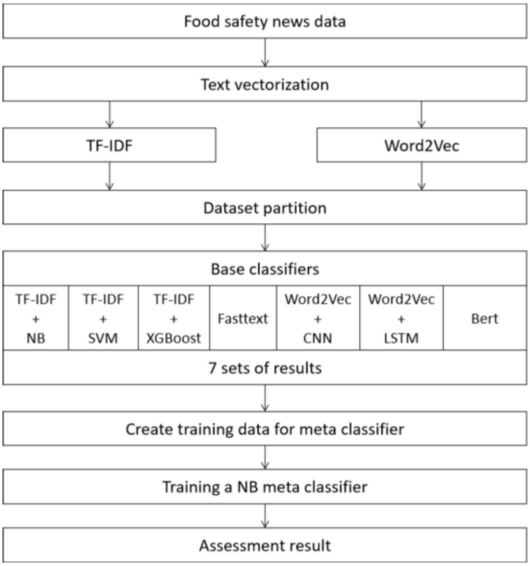

Figure 1. Ensemble classifier for news impact assessment.

2.1. Base classifier

In this research, we use the following classifiers as the base classifiers.

1. Naive Bayesian

Naive Bayesian (NB) is a probability-based machine learning algorithm. Its principle lies in Bayes theorem and the conditional independence assumption of features. When doing classification, NB calculates the probability of each feature corresponding to each class and then obtains the probability of a sample belonging to a certain class. For food safety news, the features are the TF-IDF weighted words in the vocabulary containing all the words in relevant news. Since NB is a widely used algorithm, many machine learning software packages have built-in implementation of it. In this paper we use the SKLEARN package to carry out the training and predicting with NB.

2. SVM

SVM is a classifier looking for the optimal hyperplane in the sample space such that the distance between the hyperplane and the support vectors is as large as possible, so as to ensure the accuracy of classification. SVM is usually used for binary classification. For classes more than two, such as the situation in this paper, a one-versus-rest way can be used to construct the SVM classifier. This way treats one class of the total *n* classes as a class, and the other *n*-1 classes as another class to separate the two. So totally *n* SVM classifiers will be constructed for an *n*-class classification problem.

3. XGBoost

XGBoost itself is an ensemble learner integrating multiple CART (Classification and Regression Tree) models based on the boosting mechanism. The training process of XGBoost is to create a series of CART trees and grow the trees by splitting the leaves representing decision variables. Each tree learns not to give the final decision but to fit the prediction error of a previous tree, and the final decision can be made by combining the predictions of all the trees. XGBoost applies regularization to the objective function to reduce the total number of tree leaves and the output scale of the leaves, so that overfitting can be avoided.

4. FastText

FastText is a simple three-layer neural network, including the input layer, hidden layer and output layer [10]. At the input layer, n-grams in the training text undergo a bucket hashing process and become the embedding vectors. This enables FastText to utilize the word order in n-grams and at the same time control the total number of n-grams. At the output layer, FastText uses a hierarchical softmax technique based on Huffman encoding, which reduces the number of classes to be predicted and thus improves classification efficiency. Since FastText itself has the ability to generate text vectors, we do not apply Word2Vec to the FastText classifier.

5. CNN

CNN is usually used for image classification. Yoon Kim proposed a variant of CNN, namely textCNN, for text classification [11]. In this paper, we use the word vectors generated by Word2Vec to replace the random word embedding used in textCNN, so as to incorporate more prior knowledge in the classification model. In textCNN, a passage is represented as a matrix whose rows are the embedding vectors of words in the passage, and convolution kernels are of the same length as the embedding vectors. To keep the integrity of word information, the convolution kernels are only allowed to move along the height direction. The convolution layer is followed by a k-max pooling layer. A fully connected softmax layer is used as the output layer.

6. LSTM

LSTM is a type of RNN. LSTM adds an input gate, a forgetting gate, an output gate and a memory unit to a RNN neuron, making the modified model capable of memorizing important information and forgetting unimportant information in a time series [12]. LSTM is very useful for modeling text as the word sequences in text represents time series signals. In this paper, a news article is transformed into a vector sequence and fed to the LSTM classifier. Each vector in the sequence represents the word in the corresponding position and such vectors are obtained from Word2Vec.

7. BERT

BERT [13] and its improvements [14] represent the state-of-the-art techniques in natural language processing. BERT is based on Transformer [15], an encoder-decoder architecture built on multi-head self-attention mechanism. BERT is structured as a multi-layer bidirectional Transformer encoder and is deliberately pre-trained with two types of tasks: masked language model and next sentence prediction. To use BERT for text classification, we directly input news text to the first layer of the BERT model per-trained with Chinese corpus. BERT will generate a vectorial representation of the text using the positional information of words and the knowledge it gained during the pre-training phase. A fully connected softmax layer is used to classify the generated text vectors, and training (often called fine-tuning) the BERT classifier is to learn the weight matrix of the softmax layer.

BERT is a heavy model such that the pre-training and fine-tuning of it require a lot of resources. To use BERT with limited computing resources, a practical way is to constrain the length of input. In this paper, we define 128 Chinese characters as the max length of input news for BERT. To keep as much information in the original news as possible, we apply automatic summarization to the news articles. The method of automatic summarization is TextRank [16].

2.2. Stacking

Stacking is a way to integrate the classification ability of the base classifiers. It uses the class labels (or class probabilities) output by the base classifiers and the real class label of a training sample to form a new training sample. With the newly formed training samples, a meta-classifier is trained to give the final prediction of the class label of an input sample. To prevent overfitting, the training data are divided into 7 sets of equal size shown as $a_1,\ldots, a_7$ in Figure 2. For each base classifier, it is trained 7 times. In the first time a_1 is used as the inner test set and the rest sets are used as the training set, in the second time a_2 is used as the inner test set and the rest sets are used as the training set, and so on. The predicted class labels for inner test set a_i by base classifier j is denoted as b_{ij}. By merging $b_{i1},\ldots, b_{i7}$ along the same test samples, we get B_i, and $B_1,\ldots, B_7$ comprise the training set for the meta-classifier. Since each base classifiers have 7 differently trained versions, but when testing the ensemble classifier, the meta-classifier needs a determined class label from each base classifier, we test the seven versions of the base classifier with the overall test set one by one, and choose the most frequently appeared class label for each test sample. In this fashion we get T_i, the test result of base classifier i, and by merging $T_1, \ldots, T_7$, we get T, the test set for the meta-classifier.

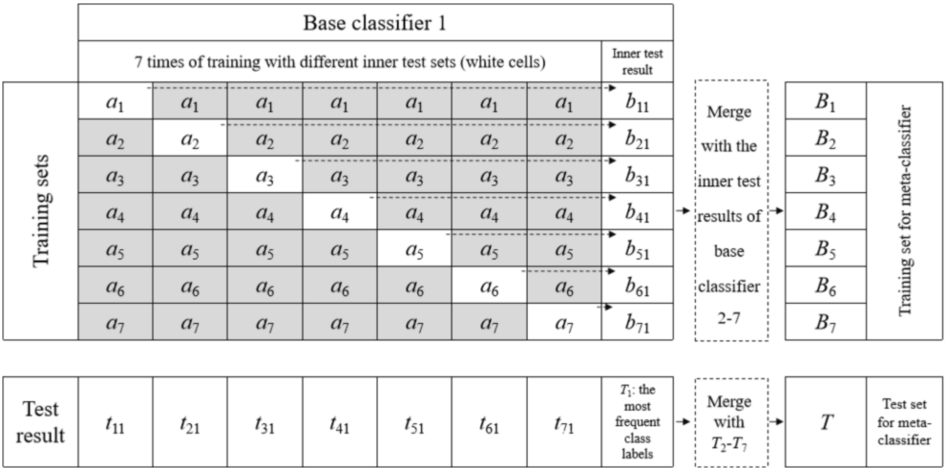
Base classifier 1
7 times of training with different inner test sets (white cells)
Inner test result
Training sets
a_1 a_2 a_3 a_4 a_5 a_6 a_7
b_{11} b_{21} b_{31} b_{41} b_{51} b_{61} b_{71}
Merge with the inner test results of base classifier 2-7
B_1 B_2 B_3 B_4 B_5 B_6 B_7
Training set for meta-classifier
Test result
t_{11} t_{21} t_{31} t_{41} t_{51} t_{61} t_{71}
T_1: the most frequent class labels
Merge with T_2-T_7
T
Test set for meta-classifier

Figure 2. Construction of training and test datasets.

3. Experiments

3.1. Data acquisition and preprocessing

The food safety news used for impact assessment experiment is acquired from the Chinese customs. Totally 21,065 news articles were collected each with a class label ranging from low impact, medium impact and high impact. The labels were assigned to the news articles by the customs officials manually. There were 10,247 low impact news articles, 10,314 medium impact news articles and 504 high impact news articles. From each kind of the news articles, a 25% fraction is randomly chosen to constitute the test set, and the rest 75% constitutes the training set. All the news articles were in Chinese because all the news written in other languages were translated by the customer officials into Chinese. After using the jieba software to segment the Chinese news, we got 88,296 distinct Chinese words, based on which the TF-IDF weights of words were calculated. The dimension of the output vector of Word2Vec (the genism version) was set to be 50.

3.2. Settings of base classifiers

Each base classifier has some hyper-parameters that control its training and predicting process. In this paper, the hyper-parameters of the base classifiers are set as Table 1.

Table 1. Hyper-parameters for base classifiers.

Base classifier	Hyper-parameters
NB	Uniform prior probabilities of classes; other parameters follow the default setting of sklearn MultinomialNB model.
SVM	Parameters follow the default setting of sklearn LinearSVC model.
XGBoost	Early stopping rounds = 10; eval_metric = "logloss"; other parameters follow the default setting of Python package xgboost.
FastText	Minimal number of word occurences = 2; other parameters follow the default setting of Python package fasttext.
CNN	Keras-based implementation of a textCNN [11]-like CNN, with a dropout layer after the embedding layer (dropout rate = 0.2); the 1D convolutional layer has 250 filters (kernal length = 3); a 3-max pooling layer follows and is followed by a flatten layer, a 50-unit dense layer and a 3-unit softmax layer; the activation function of the convolutional layer and the dense layer is ReLU; input length = 1000, batch size = 256, epochs = 5.
LSTM	Keras-based implementation of LSTM; the embedding layer is connected to a LSTM layer with 200 neurons, where a 0.2 dropout rate of the input and recurrent state is applied; following the LSTM layer is a dropout layer (dropout rate = 0.2), a 64-unit dense layer (ReLU activation function) and a 3-unit softmax layer; input length = 1000, batch size = 128, epochs = 5, Adam optimizer, learning rate = 0.01.
BERT	Chinese pre-trained model, L=12, H=768, A=12; batch size = 32, epochs = 5, learning rate = 2e-5; input length =128, input text is summarized using the "TextRank for sentence extraction" method proposed in [16].

3.3. Experiment result

We use accuracy, precision, recall and F1 score to evaluate the performance of news impact level assessment. These indexes are established based on the following concepts:

- True positive (TP) - predicted to be positive and true label is positive
- False positive (FP) - predicted to be positive but true label is negative
- True negative (TN) - predicted to be negative and true label is negative
- False negative (FN) - predicted to be negative but true label is positive

Then the abovementioned performance indexes are calculated as:

Accuracy = (TP+TN) / (TP+TN+FP+FN)
Precision = TP / (TP+FP)
Recall = TP / (TP+FN)
F1 = 2 * Precision * Recall / (Precision + Recall)

For n classes where $n > 2$, each class is chosen for calculating its own TP, FP, TN and FN, during which phase all the other classes are treated as the negative class. Then the TP, FP, TN and FN for each class is added respectively for computing the overall accuracy, precision, recall and F1 score. In this way we get the final performance of the proposed ensemble classifier, which is accuracy = 0.8062, precision = 0.8071, recall = 0.8062 and F1 = 0.8052. Figure 3 shows the confusion matrix of the ensemble classifier, where label 0, 1, 2 correspond to low impact, medium impact and high impact class respectively.

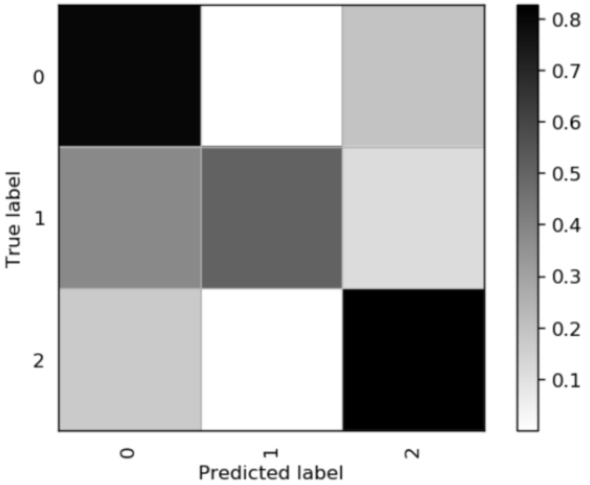

Figure 3. Normalized confusion matrix of news impact level prediction.

4. Comparative study

We carried out comparative experiments to verify the advantage of the proposed method versus existing methods. We first compared the ensemble classifier with the seven base classifiers, and then compared it with four classic classifiers, saying, the original version of NB, SVM, CNN and LSTM without using TF-IDF weights or pre-trained word vectors. The result of the comparison is shown in Table 2.

Table 2. Results of different classifiers.

Model/Index			Accuracy	Precision	Recall	F1
Classic	NB	One-hot	0.7312	0.7652	0.7312	0.7321
	SVM	One-hot	0.7532	0.7563	0.7532	0.7535
	CNN	Random embedding	0.7782	0.7882	0.7782	0.7806
	LSTM	Random embedding	0.7845	0.7852	0.7845	0.7848
Base classifier	NB	TF-IDF	0.7587	0.7615	0.7587	0.7592
	SVM	TF-IDF	0.7968	0.7990	0.7968	0.7972
	XGBoost	TF-IDF	0.7902	0.7918	0.7902	0.7905
	FastText	-	0.8027	0.8037	0.8027	0.8030
	CNN	Word2Vec	0.7870	0.7907	0.7870	0.7879
	LSTM	Word2Vec	0.8018	0.8064	0.8018	0.8024
	BERT	Auto summarization	0.7919	0.7934	0.7919	0.7921
Ensemble classifier		TF-IDF + Word2Vec	0.8062	0.8071	0.8062	0.8052

From the results in Table 2 we can see that the proposed ensemble classifier achieves the highest scores regarding all the performance indexes. The second runner is FastText, which scores 0.35% lower than the ensemble learner in terms of accuracy. LSTM with Word2Vec embedding performs similarly with FastText, but the version of LSTM with random embedding performs much poorer. By comparing NB, SVM, CNN and LSTM between their classic versions and base classifier versions, it can be seen that using information and knowledge extracted from the corpus can improve the performance of the classifiers. For BERT, which uses the most prior knowledge and represents the state-of-the-art of natural language understanding, it scores below the ensemble classifier, FastText, LSTM(base classifier version) and SVM(base classifier version), but scores above all the other classifiers. This is probably due to the shortened input of BERT.

5. Conclusion

In this paper we propose a news impact assessment method based on ensemble learning. The ensemble learner integrates NB, SVM, XGBoost, FastText, CNN, LSTM and BERT-based classifiers to classify food safety news into the low impact, medium impact and high impact categories. With word weights and word vectors learned from the corpus of news articles, the performances of the base classifiers are improved and the final ensemble learner achieves higher accuracy than many classic machine learning models. The proposed method realizes end to end assessment of news impact and can be used in the Chinese customs to improve the efficiency of food safety news assessment. It promotes the transdisciplinary research in artificial intelligence and risk management in the importation field. We also discovers that if the user is willing to sacrifice a little accuracy, then the FastText classifier can be used to predict the impact level of news, which requires a much smaller set of resources.

Acknowledgement

This work is sponsored by National Natural Science Foundation of China (71601113, 71602114), Shanghai Science & Technology Committee Research Project (17040501700, 18DZ1206802) and Shanghai Sailing Program (17YF1407700).

References

[1] B. Song, W. Yan and T.J. Zhang, Cross-border e-commerce commodity risk assessment using text mining and fuzzy rule-based reasoning, *Advanced Engineering Informatics*, 2019, Vol. 40, pp. 69–80.

[2] S.S. Fu, D. Zhang, J. Montewka, E. Zio and X.P. Yan, A quantitative approach for risk assessment of a ship stuck in ice in Arctic waters, Safety Science, 2018, Vol. 107, pp. 145–154.

[3] A. Kokangul, U. Polat and C. Dagsuyu, A new approximation for risk assessment using the AHP and Fine Kinney methodologies, Safety Science, 2017, Vol. 91, pp. 24–32.

[4] C.J. Su and Y.A. Chen, Risk assessment for global supplier selection using text mining, *Computers & Electrical Engineering*, 2018, Vol. 68, pp. 140–155.

[5] V. L. Duy, M. James, K. C. Kirkby and S. Joel, Risk prediction using natural language processing of electronic mental health records in an inpatient forensic psychiatry setting, *Journal of Biomedical Informatics*, 2018, Vol. 86, pp. 49–58.

[6] A. I. Kadhim, Survey on supervised machine learning techniques for automatic text classification. *Artificial Intelligence Review*, 2019, Vol. 52, pp. 273–292.

[7] R. A. Stein, P. A. Jaques and J. F. Valiati, An analysis of hierarchical text classification using word embeddings, *Information Sciences*, 2019, Vol. 471, pp. 216–232.

[8] T. Mikolov, K. Chen, G. Corrado and J. Dean, Efficient estimation of word representations in vector space, *International Conference on Learning Representations*, 2013, https://arxiv.org/abs/1301.3781, Accessed July 1, 2020.

[9] Z. H. Kilimci, and S. Akyokus, Deep learning- and word embedding-based heterogeneous classifier ensembles for text classification, *Complexity*, 2018, pp. 1-10.

[10] A. Joulin, E. Grave, P. Bojanowski and T. Mikolov, Bag of tricks for efficient text classification, *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume 2, Short Papers*, 2017, https://www.aclweb.org/anthology/E17-2068, Accessed July 1, 2020.

[11] Y. Kim, Convolutional neural networks for sentence classification, *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, 2014, https://www.aclweb.org/anthology/D14-1181, Accessed July 1, 2020.

[12] J. Chung, C. Gulcehre, K. H. Cho and J. Bengio, Empirical evaluation of gated recurrent neural networks on sequence modeling, *NIPS 2014 Workshop on Deep Learning*, December 2014, https://arxiv.org/abs/1412.3555, Accessed July 1, 2020.

[13] J. Devlin, M. W. Chang, K. Lee and K. Totanova, BERT: pre-training of deep bidirectional transformers for language understanding, *Proceedings of NAACL-HLT 2019*, pp. 4171–4186, https://www.aclweb.org/anthology/N19-1423.pdf, Accessed July 1, 2020.

[14] Y. Sun, S. Wang, Y. Li, X. Jiang, M. Sun and Q. Liu, ERNIE: Enhanced Representation through Knowledge Integration, *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, 2019, https://www.aclweb.org/anthology/P19-1139, Accessed July 1, 2020.

[15] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A.N. Gomez, L. Kaiser, I. Polosukhin, Attention Is All You Need. 31st Conference on Neural Information Processing Systems (NIPS 2017), 2017, https://papers.nips.cc/paper/7181-attention-is-all-you-need.pdf, Accessed July 1, 2020.

[16] R. Mihalcea and P. Tarau, Textrank: Bringing order into texts. In: *Proceedings of EMNLP*, Barcelona, Spain, July 2004, pp. 404–411.