

FedCSIS_2023_paper_3561.pdf

BERT-CLSTM model for the classification of Moroccan commercial courts verdicts

Taoufiq El Moussaoui

0000-0003-4879-7111

LISAC Laboratory, Faculty of Sciences Dhar El Mahraz
Sidi Mohamed Ben Abdellah University
Fez, Morocco

Email: taoufiq.elmoussaoui@ucmba.ac.ma

Loqman Chakir

0000-0002-8261-9370

LISAC Laboratory, Faculty of Sciences Dhar El Mahraz
Sidi Mohamed Ben Abdellah University
Fez, Morocco

Email: loqman.chakir@usmba.ac.ma

Abstract—Artificial intelligence (AI) seeks to emulate the human mind and solve complex problems by learning from available data. For many complex tasks, AI already surpasses humans in terms of accuracy, speed, and cost. Over the last decade, AI has grown increasingly advanced, and numerous industries have incorporated AI programs into their operations. The use of AI is finally beginning to permeate the legal field as well, bringing positive changes to the practice of law. The classification of commercial court verdicts has become increasingly challenging as the number of judgments issued by the courts daily has increased. In this work, we present a BERT-CLSTM model to classify verdicts of Moroccan commercial courts. By adding CLSTM to the task-specific layers of the BERT model, our model can get information on important fragments in the text. In addition, we input the representation along with the output of the BERT into the transformer encoder to take advantage of the self-attention mechanism and finally get the representation of the whole text through the transformer. The proposed model outperformed the compared baselines and achieved good results by getting an F-measure value of 93.61%.

I. INTRODUCTION

For many years, people have recognized the importance of process automation, and with nowadays technologies, it has become achievable. This automation optimizes processing time and reduces operational costs. Text mining [1] is one of the most important approaches for analyzing these massive volumes of textual data. Also, it is used to discover previously unknown relationships and propose solutions to aid decision-making. Many technologies are utilized in the text mining process to attain these aims. Text summarization, translation, classification, and information extraction are a few examples. This paper's content is limited to text classification.

Text classification is a machine-learning task that assigns a document to one or more predetermined categories based on its content. It is a key problem in natural language processing, with diverse applications such as sentiment analysis, email routing, offensive language detection, spam filtering, news classification, and language identification. Despite the advances made in text categorization performance, there is still significant potential for improvement, particularly in the Arabic language.

According to Internet World Stats, Arabic is the fourth most common language online, with over 225k users, representing

5.2% of all Internet users as of April 2019. They also reveal that it has experienced the largest user growth rate in the previous 19 years, reaching 8,917.3%. Arabic NLP is still a challenging task due to the Arabic language's richness, complexity, and complicated morphology. Arabic features are sorted in abundance aspects compared to other languages. There are several forms of grammatical, variations of synonyms word, and numerous meanings of words which differ based on factors such as the order of the word.

Traditional text classification methods use sparse vocabulary features to represent documents and treat words as the smallest unit. Documents represented by such approaches typically have high dimensionality and sparse data, so the classification accuracy is low. Later, with the rise of distributed representation, the usage of high-dimensional dense vector representation documents such as the word2vec or Glove models gradually becomes mainstream. The word vectors trained using this type of method represent the contextual semantic information of the text. Recently, with the emergence of deep learning, more researchers use deep learning neural networks for text classification, such as convolutional neural networks (CNN) and recurrent neural networks (RNN). The method using deep learning for text classification involves feeding text into a deep network to generate a representation of the text and then feeding the text representation into the softmax function to calculate the probability of each category. CNN-based models [2], [3], [4] may generate text representations with local information, whereas RNN-based models [5], [6] generate text representations with long-term information.

Nowadays, the process of classifying verdicts of Moroccan commercial courts is done manually. Court staff read each verdict and classify it into a predefined category. This process has many drawbacks. First, the processing time is long. Second, court staff may make mistakes while filing the document, and third, the confidentiality of citizens' data is not respected. Based on the disadvantages of the current system, we propose a BERT-CLSTM model that can provide the same service in a short time. The proposed model combines the advantages of CNN and RNN.

II. RELATED WORKS

Over the past few years, several researchers have addressed the issue of automatic categorization of legal data and the exploitation of these huge amounts of court-generated data to assist in decision-making. The study presented in [7] aims to classify arabic news articles based on their vocabulary features. Firstly, they used classifiers that were compatible with multi-labeling tasks such as Logistic Regression and XGBoost. XGBoost gave higher accuracy, scoring 84.7%, while Logistic Regression scored 81.3%. Secondly, ten neural networks were constructed (CNN, CLSTM, LSTM, BiLSTM, GRU, CGRU, BiGRU, HANGRU, CRF-BiLSTM, and HANLSTM). CGRU proved to be the best multi-labeling classifier scoring accuracy of 94.85%, higher than the rest of the classifiers.

Research [8] introduces AraLegal-BERT, a bidirectional encoder Transformer-based model that has been thoroughly tested and carefully optimized to amplify the impact of NLP-driven solution concerning legal documents. They fine-tuned AraLegal-BERT and evaluated it against three BERT variations for the Arabic language. The results show that the base version of AraLegal-BERT achieves better accuracy than the general and original BERT over the Legal text.

El-Alami et al [9] propose an Arabic text classification method based on Bag of Concepts and deep Autoencoder representations. It incorporates explicit semantics relying on Arabic WordNet and exploits Chi-Square measures to select the most informative features. To produce a high-level representation, they applied successive stacks of restricted Boltzmann Machines (RBMs). Experiments showed that using the Autoencoder as a text representation model combined with Chi-Square and classifier outperformed state-of-the-art techniques.

Haz et al [10] evaluated Arabic user comments on Twitter using a common form of Recurrent Neural Network (RNN), Long Short-Term Memory (LSTM). Experiments revealed that LSTM outperforms standard approaches in terms of accuracy while requiring less parameter computation, less working time, and more efficiency.

Alhawarat and Aseeri [11] implemented a CNN multi-kernel architecture with word embedding and specifically n-gram to classify Arabic documents of news, and they named the model a Superior Arabic Text Categorization Deep Model (SATCDM). Regarding the current studies on Arabic text classification, their approach achieves very high precision using 15 of the publicly available datasets.

Galal et al [12] concentrated on classifying Arabic Text using a convolution neural network (CNN), as it achieved an excellent result in various processes of natural language (NLP). They implemented GStem a new algorithm focused on extra Arabic letters and word embedding distances to group related Arabic words. Their studies have shown that it improves the accuracy of the CNN model by using GStem as a preprocessing stage since the number of separate terms has decreased.

Boukil et al [13] suggest a technique for categorizing Arabic data sets. They used an Arabic stemming algorithm to

isolate, choose and decrease features. Then, they used Term Frequency Inverse Document Frequency (TFIDF) for feature weighting. With the CNN model and other standard machine learning methods, they analyzed their dataset as a benchmark. They argued that the CNN model performs well on the Arabic text classification challenge. Standard methods, such as SVM, do not do as well as the CNN model at the stage where the dataset is large and big.

Al-khuryji and Sameh [14] proposed a new method using kernel naive Bayes for Arabic text classification. At first, they preprocessed documents such as tokenizing the Arabic words then removing the stop word and using word stemming. They used Term Frequency and Inverse Text Frequency (TF-IDF) as feature extraction, and they transformed those terms into vectors. Third, to solve the non-linearity issue of Arabic text classification, they suggested a successful solution based on the Kernel Naive Bayes (KNB) classifier. Experimental findings on the collected dataset revealed that their methodology regarding precision and time against other baseline classifiers obtained excellent results from the proposed classifier.

III. RESEARCH METHODOLOGY

In this section, we present the corpus created to train and evaluate our model, also the data preprocessing process, and finally, we explain the model architecture.

A. Dataset

The dataset created to train and evaluate our proposed classifier is a collection of Arabic verdicts issued from the Moroccan commercial courts. It consists of 2821 documents and 66900015 words. The average document size is 23715 words. Documents are categorized under four main classes: Unfair competition, Arbitration, Insurance and Commercial lease.

The dataset was divided into two sections: training dataset and testing dataset. The training dataset represents 75% of each class and it allows us to build the model, while the testing dataset represents the remaining 25% and it helps us to verify the performance of our model. Table I presents the distribution of documents in each class.

B. Data preprocessing

The first step in our preprocessing process was removing stop words, removing sign characters, punctuation by applying basic functions. Then, we execute the stemming algorithm, which transforms all words to their stems.

C. Model

The proposed BERT-CLSTM model has two characteristics. One is to use the CLSTM to transform the task-specific layer of the BERT so that our model can obtain the local and long-term representation of the text. The other is that after the CLSTM layer, we feed the representation along with the output E of the BERT into the transformer encoder. This allows the use of the self-attention mechanism to make the final text representation focus on important segments of the text. The architecture of the BERT-CLSTM model is shown in figure 1.

TABLE I
DISTRIBUTION OF DOCUMENTS IN EACH CLASS.

	Unfair competition	Arbitration	Insurance	Commercial lease	Total
Train	523	509	511	557	2100
Test	175	163	183	200	721
Total	698	672	694	757	2821

1) **CLSTM encoder**: In order to make the representation of text focus on the information in the text. We use a convolution filter to extract features from T. Assume that the size of the convolution window is $1 \times k$, the output of the CLSTM encoder is:

$$P = CLSTM_k(T) \quad (1)$$

P is $\{P_1, P_2, P_3, \dots, P_r\}$ and $r = mk + 1$. Through convolution operation, we can obtain the representation of sentences in windows with size K. For example, P_r is the representation of $T_{(r-1)}, T_r, T_{(r+1)}$.

2) **Transformer encoder**: Similar to multi-layer transformer encoder, to integrate information in P, we adopt transformer encoder to map the local representation P into the representation of whole text. The input of the transformer encoder is $\{C, P_1, P_2, P_3, \dots, P_r\}$, and we use C' which corresponds to C as the representation of the whole text.

3) **Output layer**: The output of the model is represented as follows:

$$y = \text{softmax}(\tanh(WC' + b)) \quad (2)$$

IV. RESULTS AND ANALYSIS

This section presents the experimental setup as well as the parameters utilized to train and test the model, after that, we detail the evaluation measures, and finally, we highlight the results, and we discuss them.

A. Experimental setup

To evaluate our classifier, we used the dataset that we created (See section III.A). Table II shows the model hyper-parameters.

TABLE II
MAJOR HYPER-PARAMETERS OF THE MODEL.

Step	Parameter	Value
BERT Embedding	Size of BERT	1024
BERT Embedding	BERT layer	Last 4
CLSTM Layer	Number of filters	128
CLSTM Layer	Filter sizes	3,4,5
CLSTM Layer	Pooling function	Max Pooling
CLSTM Layer	Dropout probability	0.5
CLSTM Layer	Number of epochs	30
Training	Optimizer	Adam
Training	Learning rate	1e-3
Training	Loss	Cross Entropy

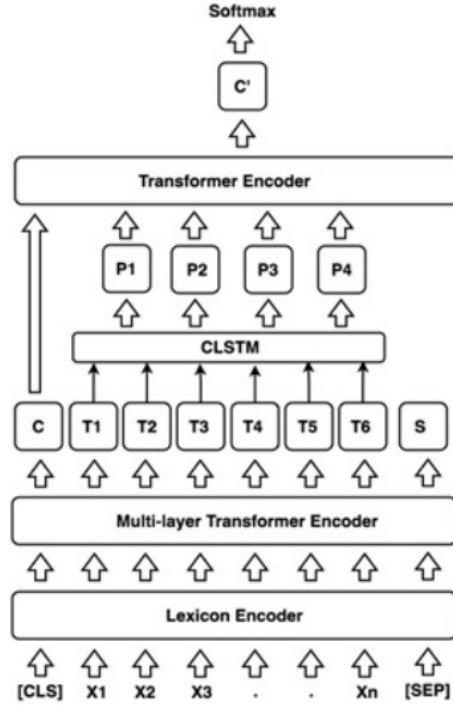


Fig. 1. Model architecture.

B. Evaluation measures

We used a variety of metrics to evaluate the performance of the model. We began by calculating the **Precision** measure, which indicates the number of documents correctly predicted by the model out of all documents predicted. The precision measure may be calculated using the equation 3:

$$\text{Precision} = \frac{\text{True_Positives}}{\text{True_Positives} + \text{False_Positives}} \quad (3)$$

Second, the **Recall** measure, which indicates the number of documents accurately predicted by the model out of the total of documents in the dataset. The recall measure can be computed using this equation 4:

$$\text{Precision} = \frac{\text{True_Positives}}{\text{True_Positives} + \text{False_Negatives}} \quad (4)$$

Third, **F1-measure** which is a metric that can be calculated based on the precision and recall using the following equation 5:

$$F1_measure = 2 * \frac{(Precision * Recall)}{(Precision + Recall)} \quad (5)$$

The last metric that we use is **Accuracy**. It is a metric used in classification problems used to tell the percentage of accurate predictions. We calculate it by dividing the number of correct predictions by the total number of predictions.

$$Accuracy = \frac{Number_of_correct_predictions}{Total_number_of_predictions} \quad (6)$$

C. Results

We applied the baseline models which are: Logistic regression (LR), Extreme Gradient Boosting (XGBoost), Naive Bayes (NB), and Random Forest (RF). Then, for deep learning models, we use the CNN algorithm with the fastText embedding and of course, our proposed model which is BERT-CLSTM. Table 3 illustrates the details of the model's evaluation findings. Precision, Recall, F1-measure, and Accuracy are denoted by the letters 'P', 'R', 'F', and 'Acc'. respectively.

TABLE III
PERFORMANCE OF OUR MODEL AGAINST THE BASELINE MODELS.

Models	P (%)	R (%)	F (%)	Acc (%)
LR + count vectors	86.07	86.07	86.07	86.07
LR + word level (TF-IDF)	86.07	86.07	86.07	86.07
LR + N-gram vectors	86.07	86.07	86.07	86.07
LR + CharLevel vectors	84.81	84.81	84.81	84.81
XGBoost + count vectors	87.34	87.34	87.34	87.34
XGBoost + word level (TF-IDF)	85.44	85.44	85.44	85.44
NB + count vectors	85.44	85.44	85.44	85.44
NB + word level (TF-IDF)	84.81	84.81	84.81	84.81
NB + N-gram vectors	87.97	87.97	87.97	87.97
NB + CharLevel vectors	79.74	79.74	79.74	79.74
RF + count vectors	84.17	84.17	84.17	84.17
RF + word level (TF-IDF)	87.97	87.97	87.97	87.97
CNN + FastText	90.98	90.52	90.75	90.68
BERT + CLSTM (our model)	93.81	93.42	93.61	93.55

D. Analysis and discussion

Table III shows that our model outperformed the baseline models and the CNN model in terms of Precision, Recall, F1-Measure and Accuracy. In term of F1-Measure, our proposed model outperforms the CNN+FastText model by 2.86, the RF+TF-IDF model and the NB+N-Gram by 5.64 as well as the XGBoost+count vectors model by 6.27 and the LR+N-Gram model by 7.54.

Figure 2 shows accuracy scores for models. In term of Accuracy, our proposed model outperforms the CNN+FastText model by 2.87, the RF+TFIDF model and the NB+N-Gram by 5.58 as well as the XGBoost+count vectors model by 6.21 and the LR+N-Gram model by 7.48.

The proposed method has a better classification effect, and the reason is that the text representation matrix has strong feature representation ability and is more representative, which can provide more category information for text classification. This also highlights the advantage of using CLSTM, which gives the representations of text with local and long-term

information, unlike the traditional methods that are based on the bag-of-words model.

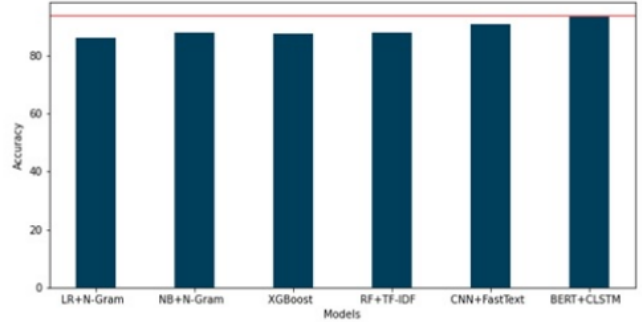


Fig. 2. Accuracy scores for models.

V. CONCLUSION

Arabic is considered one of the most difficult languages to process, due to its high morphological ambiguity, writing style, and lack of capitalization. Therefore, every NLP task involving this language requires a lot of feature engineering and preprocessing. In this paper, we present our model that classifies the Moroccan commercial court verdicts into four categories. The model outperformed the compared baselines and achieved good results by getting an F-measure value of 93.61%.

ACKNOWLEDGMENT

This work was supported by the Ministry of Higher Education, Scientific Research and Innovation, the Digital Development Agency (DDA) and the CNRST of Morocco [Alkhawarizmi/2020/36].

REFERENCES

- [1] C. Blake, "Text Mining," *Annual review of information science and technology*, vol. 45, 2011, pp. 121–155.
- [2] X. Zhang, J. Zhao, Y. LeCun, "Character-level convolutional networks for text classification," *Advances in Neural Information Processing Systems*, 113.
- [3] A. Conneau, H. Schwenk, L. Barrault, Y. LeCun, "Very deep convolutional networks for text classification," *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics*, vol. 1, 2017, pp. 1107–1116.
- [4] Y. Kim, "Convolutional neural networks for sentence classification," *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, vol. 1, 2014, pp. 1746–1751.
- [5] K. Sheng Tai, R. Socher, C.D. Manning, "Improved semantic representations from tree-structured long short-term memory networks," *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing*, vol. 1, 2015, pp. 1556–1566.
- [6] M. Huang, Q. Qian, X. Zhu, "Encoding syntactic knowledge in neural networks for sentiment classification," *ACM Transactions on Information Systems*, vol. 35, 2017, pp. 1–27.
- [7] H. Alfai, L. Al Qadi, A. Elnagar, "Arabic text classification: the need for multi-labeling systems," *Neural Computing and Applications*, vol. 34, 2022, pp. 1135–1159.
- [8] M. AL-Qurishi, S. AlQaseemi, R. Soussi, "AraLegal-BERT: A pretrained language model for Arabic Legal text," *Proceedings of the Natural Language Processing Workshop 2022*, 2022, pp. 338–344.

- [9] F. El-Alami, A. El Mahdaoui, S.O. El Alaoui, N. En-Nahnahi, "A deep autoencoder-based representation for Arabic text categorization," *Journal of Information and Communication Technology*, vol. 3, 2020, pp. 381–312.
- [10] W.H.G. Gwad, I.M.I. Ismael, Y. Gultepe, "Twitter Sentiment Analysis Classification in the Arabic Language using Long Short-Term Memory Neural Networks," *International Journal of Engineering and Advanced Technology*, vol. 9, 2020, pp. 235–239.
- [11] M. Alhawarat, A.O. Aseeri, "A Superior Arabic Text Categorization Deep Model (S 21 DM)," *IEEE Access*, vol. 8, 2020, pp. 24653–24661.
- [12] M. Galal, M.M. Madbouly, A. El-Zoghby, "Classifying Arabic text using deep learning," *Journal of Theoretical and Applied Information Technology*, vol. 97, 2019, pp. 3412–3422.
- [13] S. Boukil, M. Biniz, F. El Adnani, L. Cherrat, A.E. El Moutaouakkil, "Arabic Text Classification Using Deep Learning Technics," *International Journal of grid and distributed computing*, vol. 11, 2018, pp. 103–114.
- [14] R. Al-khuryaji and A. Sameh, "An Effective Arabic Text Classification Approach Based on Kernel Naïve Bayes Classifier," *International Journal of Artificial Intelligence and Applications*, vol. 8, 2017, pp. 01–10.

59%

SIMILARITY INDEX

PRIMARY SOURCES

- | | | |
|---|---|-----------------|
| 1 | pdfs.semanticscholar.org
Internet | 399 words — 12% |
| 2 | dokumen.pub
Internet | 345 words — 11% |
| 3 | Alaa Eddine El Moussaoui, Taoufiq El Moussaoui, Brahim Benbba, Anicia Jaegler, Zineb El Andaloussi. "Understanding the Choice of Collection & Delivery Point by the E- Consumer via a Machine Learning Model: Moroccan Case Study", Procedia Computer Science, 2022
Crossref | 207 words — 6% |
| 4 | www.researchgate.net
Internet | 143 words — 4% |
| 5 | link.springer.com
Internet | 100 words — 3% |
| 6 | aclanthology.org
Internet | 71 words — 2% |
| 7 | Hai Huan, Zelin Guo, Tingting Cai, Zichen He. "A text classification method based on a convolutional and bidirectional long short-term memory model", Connection Science, 2022
Crossref | 68 words — 2% |

8	www.mdpi.com Internet	47 words — 1%
9	Sireesha Jasti, G.V.S. Raj Kumar. "Adaptive Rider Feedback Artificial Tree Optimization-Based Deep Neuro-Fuzzy Network for Classification of Sentiment Grade", Journal of Telecommunications and Information Technology, 2023 Crossref	40 words — 1%
10	www.law.georgetown.edu Internet	40 words — 1%
11	www.mecs-press.org Internet	35 words — 1%
12	avesis.atauni.edu.tr Internet	34 words — 1%
13	www.thieme-connect.com Internet	30 words — 1%
14	Siham Akil, Sara Sekkate, Abdellah Adib. "Chapter 9 Combined Feature Selection and Rule Extraction for Credit Applicant Classification", Springer Science and Business Media LLC, 2023 Crossref	29 words — 1%
15	aircconline.com Internet	27 words — 1%
16	downloads.hindawi.com Internet	26 words — 1%
17	Najm Alotaibi, Badriyya B. Al-onazi, Mohamed K. Nour, Abdullah Mohamed et al. "Political Optimizer with Probabilistic Neural Network-Based Arabic Comparative	25 words — 1%

Opinion Mining", Intelligent Automation & Soft Computing, 2023

Crossref

-
- | | | |
|----|---|---------------|
| 18 | e-journal.uum.edu.my
<small>Internet</small> | 24 words — 1% |
|----|---|---------------|
-
- | | | |
|----|---|---------------|
| 19 | www.politesi.polimi.it
<small>Internet</small> | 23 words — 1% |
|----|---|---------------|
-
- | | | |
|----|---|---------------|
| 20 | www.stmik-budidarma.ac.id
<small>Internet</small> | 23 words — 1% |
|----|---|---------------|
-
- | | | |
|----|---|---------------|
| 21 | igsr.alexu.edu.eg
<small>Internet</small> | 22 words — 1% |
|----|---|---------------|
-
- | | | |
|----|--|---------------|
| 22 | Murat Aydoğan, Ali Karci. "Improving the accuracy using pre-trained word embeddings on deep neural networks for Turkish text classification", Physica A: Statistical Mechanics and its Applications, 2020
<small>Crossref</small> | 20 words — 1% |
|----|--|---------------|
-
- | | | |
|----|---|---------------|
| 23 | "Digital Technologies and Applications", Springer Science and Business Media LLC, 2023
<small>Crossref</small> | 19 words — 1% |
|----|---|---------------|
-
- | | | |
|----|--|---------------|
| 24 | Hanane Elfaik, El Habib Nfaoui. "Deep Bidirectional LSTM Network Learning-Based Sentiment Analysis for Arabic Text", Journal of Intelligent Systems, 2020
<small>Crossref</small> | 17 words — 1% |
|----|--|---------------|
-
- | | | |
|----|--|-----------------|
| 25 | "Intelligence Science and Big Data Engineering. Big Data and Machine Learning", Springer Science and Business Media LLC, 2019
<small>Crossref</small> | 13 words — < 1% |
|----|--|-----------------|
-
- | | | |
|----|---|-----------------|
| 26 | article.nadiapub.com
<small>Internet</small> | 13 words — < 1% |
|----|---|-----------------|
-

27	mafiadoc.com Internet	13 words — < 1%
28	dl.acm.org Internet	11 words — < 1%
29	Sridhar R. Kundur, Daniel Raviv. "Active vision-based control schemes for autonomous navigation tasks", Pattern Recognition, 2000 Crossref	10 words — < 1%
30	ikee.lib.auth.gr Internet	10 words — < 1%
31	dspace.adu.ac.ae Internet	9 words — < 1%
32	ebin.pub Internet	9 words — < 1%
33	qspace.qu.edu.qa Internet	9 words — < 1%
34	"Artificial Intelligence and Smart Environment", Springer Science and Business Media LLC, 2023 Crossref	8 words — < 1%
35	"Natural Language Understanding and Intelligent Applications", Springer Science and Business Media LLC, 2016 Crossref	6 words — < 1%