ISSN 1330-3651 (Print), ISSN 1848-6339 (Online)
https://doi.org/10.17559/TV-20231126001151
Original scientific paper

Fake News Detection Using Deep Neuro-Fuzzy Network

Ning PAN

Abstract: In this study, we introduce an innovative network architecture that synergizes fuzzy neural networks with positional self-attention mechanisms to enhance fake news detection. This approach effectively addresses emerging challenges posed by new fake news technologies, aiming to bolster detection accuracy, protect public interests, and support credible media development. By integrating diverse information sources, including textual content and semantic nuances, our model excels in processing ambiguous data and discerning subtle variances in news authenticity. The utilization of fuzzy neural networks allows for adept handling of uncertain information, while positional self-attention coding proficiently identifies the significance of different textual elements, offering a nuanced analysis of news veracity. Our extensive experiments on two datasets reveal a substantial improvement in detection accuracy, with the model achieving an accuracy increase of over 15% compared to traditional methods. This work not only demonstrates a methodological advancement in tackling fake news but also contributes significantly to upholding social integrity and public trust.

Keywords: artificial intelligence; fake news; fuzzy rules; positional coding

1 INTRODUCTION

Fake news is news content that intentionally publishes, disseminates or promotes false information. Such false information may be misleading, misleading or completely fabricated, and is intended to mislead readers, viewers or listeners. The existence of false news may have serious implications for society and individuals, including undermining public trust, manipulating public opinion and destabilizing society. News is valued for its truthfulness and objectivity [1]. However, with the rapid development of the Internet, artificial intelligence and other new technologies, the cost of counterfeiting has become extremely low. In the age of artificial intelligence, although the main body of false information production is still human, the role of machines is becoming more and more prominent [2]. In particular, the introduction of ChatGPT at the end of 2022 will have a disruptive impact on the news content production model and the news industry. Some fake news and Internet rumours involving sensitive topics spread very fast, causing great social harm [3]. Fake news detection is an important task aimed at identifying and filtering out fake news in order to protect the public from false information [1]. Fake news detection typically involves the use of natural language processing (NLP) and machine learning techniques to analyse the content, semantics, and context of news texts to determine their authenticity and credibility [4]. Common fake news detection methods include machine learning methods based on textual features, deep learning-based methods (e.g., convolutional neural networks and recurrent neural networks), and methods based on social media data. These methods can automatically identify fake news by training models and provide users with accurate news evaluation and classification. The goal of Fake News Detection is to provide a reliable tool to help users identify and avoid the effects of false information and to promote public access to and dissemination of authentic and credible news [1-4]. Early automatic fake news detection methods focused on extracting features from news articles for classification, such as article sentiment features and topic features. However, in today's high-speed development of big data and artificial intelligence, the discovery, verification and processing of fake news relying entirely on manual labour can no longer meet the needs of the situation, and machine learning based fake news detection methods have been continuously developed [1-4]. Traditional machine learning fake news detection methods utilize supervised classifiers such as decision trees, random forests and support vector machines (SVMs) [5-7]. Although these methods can improve the accuracy of fake news detection to some extent, they mainly rely on time-consuming and laborious hand-crafted features [1-4]. Recently, deep learning-based methods have gradually taken over the mainstream, employing variants of recurrent neural networks that can sequentially process rumour propagation sequences and obtain propagation features for fake news [7-9]. To improve the performance even further, many researchers have extracted more distant dependency features through the Transformer model. Transformer-based Bidirectional Encoder Representations from Transformers is one of the most successful pre-trained language models for text classification [1-4]. Fake news detection is a challenging task, mainly composed of the following challenges:1) Diversity and complexity: Fake news comes in a variety of forms and contents, including misleading headlines, false statements of fact, and falsified evidence. Producers of fake news also constantly change their strategies and tactics, making them more difficult to detect and recognize. 2) Data availability and quality: Obtaining large-scale datasets of both real and fake news is a challenge. The amount of fake news is relatively small, while data for real news is more readily available. In addition, there may be noise, mislabelled or inaccurate information in the data, which may affect the performance of the model. 3) Context and Semantic Understanding: the detection of fake news needs to consider the context and semantic information of the text. Fake news may mislead readers by using ambiguous language, implicit information or ambiguity. Therefore, the model needs to have the ability to understand context and semantics to accurately identify false information. 4) Rapid dissemination and timeliness: False news tends to spread rapidly, especially on platforms such as social media. Therefore, fake news detection needs to be real-time and responsive in order to identify and respond to the spread of false information in a timely manner. 5) Adversarial attacks: those who create fake news may consciously try to circumvent the detection algorithms by modifying the text, using antagonistic samples, or other technological means to deceive the model. This makes fake news detection challenged by adversarial attacks and requires continuous improvement and updating of the model to cope with these attacks. In order to address the above challenges, this paper proposes a network architecture that combines the self-attention mechanism and fuzzy neural networks for fake news detection. Specifically, text data is first taken as input and the text is converted into a vector representation using word embedding techniques. Then a fuzzy neural network layer is constructed, and each hidden layer can use fuzzy sets and fuzzy rules to model and process fuzzy inputs and outputs. A self-attention mechanism layer is introduced after each implicit layer of the fuzzy neural network. The self-attention mechanism learns the relationships between different positions in the text and calculates the attention weights for each position based on these relationships. These attention weights can be used to weight the output of the fuzzy neural network. An output layer is connected after the last hidden layer to get the classification of fake news. The key contributions of our work are as follows: Innovative Integration: We propose a unique integration of fuzzy logic with neural networks, enhancing the model's ability to handle ambiguous and uncertain information typically found in fake news. Advanced Positional Analysis: By incorporating positional self-attention mechanisms, our model gains a more nuanced understanding of text context, enabling more accurate identification of fake news. Enhanced Accuracy: Preliminary results indicate that our approach significantly improves the accuracy of fake news detection compared to existing methods. Broad Applicability: The methodology developed in this study can be applied to diverse data sets, demonstrating its versatility in different contexts of fake news detection. To guide readers through our exploration of advanced fake news detection: Introduction: Outlining the research problem and its importance. Literature Review: Contextualizing our work within existing research. Methodology: Detailing our innovative approach combining fuzzy neural networks with positional self-attention. Results: Presenting our findings and comparing them with existing benchmarks. Discussion: Analyzing the implications and limitations of our approach. Conclusion: Summarizing key contributions and suggesting future research directions.

2 RELATED WORKS

2.1 Machine Learning Based Detection

Traditional machine learning methods usually rely on feature engineering. Feature engineering first cleans and preprocesses the raw text data, including removing special characters, punctuation marks, stop words, etc., and performs text normalization operations such as word stemming or word shape reduction [1-4]. Then meaningful features are extracted from the pre-processed text. Common features include word frequency, TF-IDF (Word Frequency-Inverse Document Frequency), bag-of-words model, n-gram model and so on [1-4]. These features can capture the semantic and structural information of the text. Finally, feature selection methods are used to select the most discriminative features. Common feature selection methods include mutual information, chi-square test, information gain, etc. By selecting the most relevant features, the performance and efficiency of the model can be improved [5, 6]. After feature extraction, a classifier needs to be designed and trained to classify the news. Commonly used classifiers include Simple Bayes, Support Vector Machines, Decision Trees and Random Forests [2]. These traditional machine learning algorithms usually use a supervised learning approach. The model uses the extracted features as inputs to classify the text as real news or fake news. For example, Nurhasanah et al. [5] applied machine learning models to classify fake news and evaluated their effectiveness by comparing the performance of SVM and Bayesian models. The used dataset contains reliable and unreliable news labels, which were feature extracted and pre-processed. Then, SVM and Bayesian models were trained and evaluated in classifying fake news. Choudhury et al. [6] proposed a genetic algorithm based fake news detection method and analysed the performance of SVM, Bayes, random forest and logistic regression classifiers in comparison on different datasets. The results show that the SVM classifier achieves high accuracy on multiple datasets, proving its effectiveness in fake news detection. Also, the study demonstrates the potential of genetic algorithm-based fake news detection algorithms. However, traditional machine learning methods have certain limitations in fake news detection, such as the limitation of feature representation and the efficiency of handling large-scale data. In recent years, the rise of deep learning methods has made significant progress in fake news detection because they can automatically learn feature representations and process large-scale data [7-10].

2.2 Deep Learning Based Detection

Deep learning based fake news detection is a method that uses deep learning techniques to identify and detect fake news [7, 8]. It determines the authenticity and credibility of news by training and predicting deep learning models on the news text. Deep learning is a branch of machine learning that simulates the working principle of neural networks in the human brain and carries out feature extraction and pattern recognition through multi-layer neural networks [9, 10]. In fake news detection, deep learning models can learn and understand the semantics and context in the news text to determine whether the news is true or not. Deep learning methods for fake news detection usually include the following steps: first, the original news text is converted into a format that can be processed by the machine learning model, for example, the text is converted into a numeric vector representation. Then an appropriate deep learning model architecture, such as recurrent neural network, long and short-term memory network, or convolutional neural network, is selected, and the model is trained and optimized. The news text is feature extracted by deep learning models to capture semantic and contextual information in the text [11-13]. Deep learning has some advantages in fake news detection, such as the ability to handle large-scale text data, automatically learn feature representations and model complex semantic relationships. However, the training and tuning of deep learning models require large amounts of data and computational resources, and may suffer from overfitting for small-scale datasets. For example, Raza et al. [7] proposed a Transformer-based self-encoder architecture that addresses the challenges of early detection and insufficient labelling data in fake news detection. In addition, the model incorporates many features of news content and social contexts into feature modelling. Zhang et al. [9] proposed a deep learning-based fast fake news detection model for dealing with the problem of fake news in online social services. Considering that news is usually a short text and can be significantly characterized by some keywords, a convolution-based neural computation framework was used to extract feature representations of the news text. However, existing deep learning models still have their limitations. Specifically, the forms and contents of fake news are diverse and constantly changing. This makes the models need to be adaptive and generalizable to cope with new forms of fake news [14]. Secondly, fake news may use misleading language and techniques, and the models need to have good semantic understanding and be able to recognize implied intentions and messages [15]. In addition, deep learning models are often considered black-box models and it is difficult to interpret their predictions, while interpretability and explain ability are important for users and decision makers. In order to address the above challenges, this paper proposes a network architecture that combines the self-attention mechanism and fuzzy neural networks for fake news detection. Deep neuro-fuzzy networks have the following advantages in fake news detection:1) Dealing with ambiguity and uncertainty: One of the characteristics of fake news is the ambiguity and uncertainty of information. Fuzzy neural networks can effectively deal with this vagueness and uncertainty by modelling and processing fuzzy inputs and outputs through fuzzy sets and fuzzy rules. This makes fuzzy neural networks better able to cope with the ambiguity and uncertainty present in fake news [14]. 2) Nonlinear modelling ability: fuzzy neural networks have strong nonlinear modelling ability and can learn and represent complex nonlinear relationships. Fake news detection often involves semantic and contextual information of the text, which is often nonlinear. Fuzzy neural networks can construct complex nonlinear mapping relationships through multiple hidden layers and nonlinear activation functions to better capture the features and patterns of fake news [15]. 3) Adaptive learning ability: fuzzy neural networks have adaptive learning ability, which can automatically adjust the weights and parameters of the network through training data. This enables the fuzzy neural network to learn and adjust according to different fake news samples to improve the performance and accuracy of the model [16]. 4) Explanatory and interpretable: the fuzzy inference system of fuzzy neural network can provide a certain degree of explanatory and interpretable. It can describe the relationship between inputs and outputs through fuzzy sets and fuzzy rules, thus making the decision-making process of the model more transparent and understandable. This is very important for fake news detection because users need to understand how the model determines whether a piece of news is fake or not [14-16].

3 METHODS
3.1 Overall Framework

To further elucidate the choice of methods in this study, it is important to understand the specific challenges posed by fake news detection. Given the complexity and evolving nature of fake news, we chose fuzzy neural networks due to their ability to handle ambiguous and uncertain information, which is a common characteristic in fake news. The inclusion of fuzzy logic allows for a more nuanced approach to processing the subtleties and uncertainties often present in the language of fake news. Additionally, we implemented positional self-attentive coding because of its effectiveness in understanding the context and significance of different parts of the text. This is crucial in fake news detection, where the position of words or phrases can significantly alter the meaning and intent of the news content. Positional self-attention helps in discerning these nuances, enhancing the model's ability to accurately identify fake news. Therefore, the integration of these methods is not arbitrary but a strategic choice aimed at addressing the specific demands of fake news detection. They complement each other to create a robust model capable of effectively identifying and analyzing the intricacies of fake news, which traditional methods might overlook. This paper proposes a network architecture that combines the self-attention mechanism and fuzzy neural networks for fake news detection, as shown in Fig. 1. Specifically, text data is first taken as input and the text is converted into a vector representation using word embedding techniques. Then a fuzzy neural network layer is constructed, and each hidden layer can use fuzzy sets and fuzzy rules to model and process fuzzy inputs and outputs. A self-attention mechanism layer is introduced after each implicit layer of the fuzzy neural network. The self-attention mechanism learns the relationships between different positions in the text and calculates the attention weights for each position based on these relationships. These attention weights can be used to weight the output of the fuzzy neural network. A binary function layer is connected after the last hidden layer to get the classification of fake news.

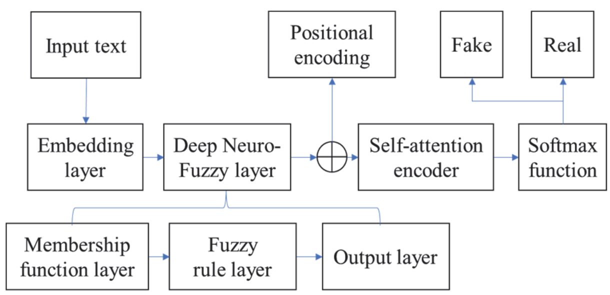

Figure 1 Overall Framework

3.2 Embedding Layer

An embedding layer is a fundamental component in deep learning models, particularly in natural language processing (NLP) tasks. It is responsible for transforming discrete input data, such as words or categorical variables, into continuous, dense vector representations called embeddings. In the context of NLP, an embedding layer maps each word or categorical feature to a low-dimensional vector representation. These vectors capture semantic and syntactic relationships between words, allowing the model to understand the contextual meaning of words in a given text. The embedding layer takes an input sequence of integers, where each integer represents a word or category. It then looks up the corresponding embedding vector for each input integer from a pre-trained embedding matrix or learns the embeddings from scratch during the training process. The output of the embedding layer is a sequence of continuous vectors, where each vector represents the embedded representation of the corresponding input word or category. The main advantage of using an embedding layer is that it enables the model to learn meaningful representations of words or categories, capturing their semantic relationships and similarities. These learned embeddings can then be used as input features for downstream tasks such as sentiment analysis, machine translation, or text classification. Assume we have a discrete input feature x, which takes values from 1 to V, where V is the total number of features. We want to map each feature x to a d-dimensional continuous vector representation, where d is the specified embedding dimension. We can use an embedding matrix E of size $V \times d$ to represent the embedding vectors for each feature. Each row of the embedding matrix corresponds to the embedding vector for a specific feature. For the input feature x, we can obtain its corresponding embedding vector by looking up the x-th row of the embedding matrix E. This can be represented by the following Eq. (1):

$$h = E\left[x\right] \tag{1}$$

where h represents the embedding vector for the feature x. During the training process, the weights of the embedding matrix E are learned and updated to better capture the semantic relationships and similarities between features.

3.3 Deep Neuro-Fuzzy Layer

Deep Neuro-Fuzzy Layer is a hybrid layer that combines neural networks and fuzzy logic. It combines the hidden layer of a neural network with the fuzzy inference of fuzzy logic to enhance the expressive and inference capabilities of the model. In traditional neural networks, neurons in the hidden layer transform and extract features from the input by means of a nonlinear activation function while in deep neuro-fuzzy layer each hidden layer neuron not only has the feature extraction ability of traditional neural network, but also has the ability of fuzzy inference.

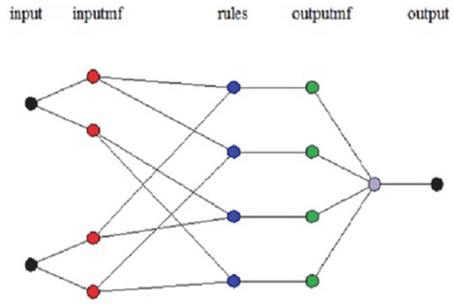

Figure 2 Deep Neuro-Fuzzy layer

The neurons of deep neuro-fuzzy layer usually consist of two parts: fuzzy sets and fuzzy rules. Fuzzy sets are used to represent the degree of fuzziness of inputs, while fuzzy rules define the relationship between inputs and outputs. The neuron learns and adjusts the weights of the fuzzy sets and fuzzy rules in order to realize fuzzy reasoning and decision making.

Fig. 2 shows the deep neuro-fuzzy layer framework. Specifically, the Embedding layer is followed by an affiliation function layer with $n \times r$ nodes, each representing a Gaussian functional type of membership function (MF). The output of the MF layer is:

$$\mu_{ij}\left(x_i\right) = \exp\left[-\frac{\left(x_i - c_{ij}\right)^2}{\sigma_{ij}^2}\right] \tag{2}$$

$$i = 1, 2, ..., n; j = 1, 2, ..., r$$

where μ_{ij} is the value of the j-th MF of x_i, c_{ij} and σ_{ij} are the center and width of the j-th MF of x_i, respectively. After that a rule layer is connected and the output is:

$$\varphi_j\left(x\right) = \prod_{i=1}^{n} \mu_{ij}\left(x_i\right) = \exp\left[-\sum_{i=1}^{n}\frac{\left(x_i - c_{ij}\right)^2}{\sigma_{ij}^2}\right] \tag{3}$$

The normalized output of this layer is.

$$h_j = \frac{\varphi_j\left(x\right)}{\sum_{k=1}^{r}\varphi_k\left(x\right)} = \frac{\exp\left[-\sum_{i=1}^{n}\frac{\left(x_i - c_{ij}\right)^2}{\sigma_{ij}^2}\right]}{\sum_{k=1}^{r}\exp\left[- -\sum_{i=1}^{n}\frac{\left(x_i - c_{ik}\right)^2}{\sigma_{ik}^2}\right]} \tag{4}$$

In which $h = [h_1, h_2, \ldots, h_r]^T$ is the normalized output vector of the rule layer. The overall output is:

$$y_q\left(x\right) = \sum_{j=1}^{r} w_{jq} h_j = \frac{\sum_{j=1}^{r} w_{jq}\exp\left[-\sum_{i=1}^{n}\frac{\left(x_i - c_{ij}\right)^2}{\sigma_{ij}^2}\right]}{\sum_{k=1}^{r}\exp\left[-\sum_{i=1}^{n}\frac{\left(x_i - c_{ik}\right)^2}{\sigma_{ik}^2}\right]} \tag{5}$$

in which w_{jq} is the weight of the jth rule point with the qth output node.

3.4 Position Based Attention Mechanism

Position-based attention mechanism is a commonly used attention mechanism in NLP. It is used to process sequential data, such as text or speech, to capture the importance of different positions in the input sequence. In the traditional attention mechanism, the attention weights are determined by calculating the similarity between each position in the input sequence and the target position. However, for long sequences or tasks with significant positional information, this computation may not adequately capture the importance of the positions. Position-based attention mechanism solves this problem by introducing position coding. Position encoding is a technique that embeds position information into the input sequence by assigning a unique vector representation to each position. In this way, the model can determine the importance of different locations by learning the weights of the position encoding. In an attentional mechanism using location coding, the attentional weights are computed considering not only the similarity between each location in the input sequence and the target location, but also the weights of the location coding. In this way, the model can better distinguish the importance of different locations and focus attention more accurately. The formula for the position-based attention mechanism is expressed as follows: suppose we have an input sequence.

$$X = \left[x_1, x_2, ..., x_n\right] \tag{6}$$

where x_i denotes the i-th element in the sequence. We wish to compute the attentional weight of each element x_i with respect to the target location. 1) Compute the attentional weights: first, we assign a position-encoding vector $PE(i)$ to each position i. The position-encoding vector can be computed using a sinusoidal vector. The position encoding vector can be computed using sine and cosine functions to capture the relationship between positions. Then, we compute the similarity score between each location i and the target location j. The similarity score can be computed using the dot product, additive attention, and the additive attention function. Similarity scores can be computed using dot product, additive attention, and scaled dot product attention. Next, we normalize the similarity scores by the softmax function to obtain the attention weights $\alpha(i, j)$. The attention weight indicates the importance of position i to the target position j. The attention weights are then normalized by the softmax function to obtain $\alpha(i, j)$. 2) Applying attentional weights: Use the attentional weights $\alpha(i, j)$ to weight and sum the input sequence X to get the weighted representation $C(j)$.

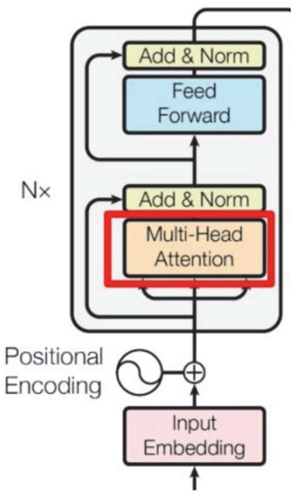

Figure 3 Positional encoding and self-attention [17]

The weighted representation $C(j)$ is the result of attentional focusing of the input sequence X on the target location j. The above formula describes the location-based attentional focusing of the input sequence X on the target location. The above formula describes the calculation process of the location-based attention mechanism. By calculating the attention weights, the model can strengthen or weaken the degree of attention to different elements in the sequence according to the relationship between different positions and the target position, so as to better capture the important information in the sequence. Fig. 3 shows the structure of positional encoding and self-attention [17-19].

3.5 Detection

After extracting the sequence features, the features are fed into the softmax function and the output of the softmax function is the final classification result. The Softmax function is a popular mathematical function commonly used for classification tasks in machine learning and deep learning. It converts a set of real numbers into a probability distribution such that each real number has a value ranging from 0 to 1 and the sum of all real numbers equals 1. Given an input vector $x = [x_1, x_2, ..., x_n]$, where x_i denotes the i-th element in the vector. The Softmax function outputs a vector $y = [y_1, y_2, ..., y_n]$ for each element. The equation for y_i is given below:

$$y_i = \exp\left(x_i\right) / \left(\sum_j \exp\left(x_j\right)\right) \tag{7}$$

in which $\exp(x)$ denotes the natural exponential function and Σ_j denotes summation over all elements.

The overall framework is to solve an optimization problem where the objective is to minimize the cross-entropy loss function and the decision variables are the parameters of all hidden layers.The solver is the adam optimizer.

4 RESULTS
4.1 Datasets

We collected two fake news detection datasets to test our proposed method. Dataset 1 contains 17903 fake news texts, of which 39% are News, 29% are politics, and 32% are other articles. Dataset 1 also contains 21192 real news articles. Dataset 2 contains 6335 fake and real news with headlines and authenticity labels.

4.2 Experimental Setups

The experimental environment for this paper is Tensorflow 2.0, python 3, and the hardware is an RTX2080 GPU with a quad-core intel i7-7700 processor. Due to GPU memory limitation, the batch size is set to 32, and the initial learning rate is 0.001 with adaptive fading processed by Adam optimizer. The evualution metrics in this paper are precise, recall, F1 and accuracy. Precision measures the proportion of samples correctly judged as positive examples by the classifier out of all samples judged as positive examples by the classifier. As shown in Eq. (8):

$$Precision = TP / \left(TP + FP\right) \tag{8}$$

where TP (True Positive) represents the number of samples correctly judged as positive by the classifier and FP (False Positive) represents the number of samples incorrectly judged as positive by the classifier. Recall measures the proportion of samples correctly judged as positive by the classifier among all actual positive samples as shown in Eq. (9):

$$Recall = TP / \left(TP + FN\right) \tag{9}$$

where FN (False Negative) represents the number of samples incorrectly judged as negative examples by the classifier. F1 Score is a combined measure of precision and recall, which is the reconciled average of precision and recall. As shown in Eq. (10):

$$F1 \text{ Score} = \frac{2 \cdot \left(Precision \cdot Recall\right)}{\left(Precision + Recall\right)} \tag{10}$$

the $F1$ Score combines Precision and Recall and allows for a comprehensive assessment of the performance of a classifier. Accuracy measures the proportion of all samples that the classifier correctly classifies.

$$Accuracy = \left(TP + TN\right) / \left(TP + TN + FP + FN\right) \tag{11}$$

where TN (True Negative) represents the number of samples correctly judged as negative examples by the classifier.

To ensure the reproducibility of our study and provide a clear understanding of the model's training process, the following details are added regarding hyperparameter settings and model training:

Embedding Layer: We used a 100 dimensional vector for word embeddings. The embeddings were initialized randomly and fine-tuned during training.

Fuzzy Neural Network Layer: The number of neurons in this layer was set to 64, with each neuron representing a fuzzy set. The membership functions were Gaussian with parameters optimized during training.

Positional Self-Attention Layer: We implemented 4 attention heads with a hidden size of 256 for each head. The dropout rate applied was 0.1 to prevent overfitting.

Learning Rate and Optimization: An initial learning rate of 0.001 was used with the Adam optimizer. The learning rate was scheduled to reduce by 10% every 5 epochs based on validation loss.

Batch Size and Epochs: The model was trained with a batch size of 32. We trained the model for a total of 30 epochs, using early stopping based on the validation set performance to prevent overfitting.

Data Preprocessing: We performed standard preprocessing steps including tokenization, removing special characters, and converting all texts to lowercase.

Data Splitting: The dataset was divided into training (70%), validation (15%), and testing (15%) sets.

Model Training: The model was trained on the training set, with validation performed at the end of each epoch to monitor performance and adjust hyperparameters if necessary.

Model Evaluation: After training, the model was evaluated on the test set to assess its performance on unseen data.

By providing these specific details, other researchers can replicate our study more easily, ensuring the reliability and validity of our findings.

4.3 Comparisons with Baselines

The baseline models most include traditional machine learning algorithms such as KNN, SVM, DT, and advanced deep learning text classification models including Bi-LSTM, Transformer encoder and TextCNN. Tab. 1 shows the settings of baseline models. We use a 10-fold cross-validation method to guarantee the robustness of the results. The specific steps are as follows:

1. randomly divide the original dataset into 10 equal-sized subsets.
2. For each subset, use it as the validation set and merge the other 9 subsets as the training set.
3. train the model using the training set and perform performance evaluation on the validation set.
4. Repeat steps 2 and 3 until each subset is used as a validation set.
5. Average the results of the 10 evaluations to obtain the final model performance evaluation.

The 10-fold cross-validation can more accurately assess the performance of the model because it uses the entire dataset for training and validation, avoiding the chance that may be introduced by a single division. It can also help detect overfitting or underfitting conditions of the model and provide a better estimate of the model's generalization ability.

Table 1 Settings of baseline models

Method	settings
KNN	- n_neighbors: 5 - weights: 'uniform' - metric: 'euclidean'
SVM	- C: 1.0 - kernel: 'rbf' - gamma: 'scale'
DT	- criterion: 'gini' - max_depth: None - min_samples_split: 2
Bi-LSTM	- units: 128 - dropout: 0.2 - recurrent_dropout: 0.2
Transformer encoder	- num_layers: 4 - d_model: 256 - num_heads: 8 - d_ff: 1024
TextCNN	- num_filters: 128 - filter_sizes: [3, 4, 5] - dropout: 0.5

Tab. 2 shows the comparations on dataset 1. The experiments on dataset 1 show that the best detection performance is for our proposed model. The best performer in traditional machine learning is DT and the best performer in deep learning is Bi-LSTM. Compared to the DT model, our proposed model improves precision by 5.23%, recall by 8.21%, F1 value by 6.04% and accuracy by 6.78%. Compared to Bi-LSTM model, our proposed model improves the precision by 0.00%, recall by 3.78%, F1 value by 2.40% and accuracy by 1.66%.

Table 2 Comparations on dataset 1

Method	*Precise*	*Recall*	*F*1	*Accuracy*
KNN	83.57%	85.12%	84.33%	83.98%
SVM	86.24%	82.76%	84.47%	84.12%
DT	84.89%	83.21%	84.04%	83.67%
Bi-LSTM	89.12%	87.45%	88.27%	88.79%
Transformer encoder	88.57%	89.12%	88.84%	88.78%
TextCNN	87.32%	88.45%	87.88%	87.79%
Proposed model	90.12%	91.23%	90.67%	90.45%

Based on the results in Tab. 3, we can see that the proposed model still has the best performance with the highest precision, recall, F1 value and accuracy on dataset 2. Compared to other methods, our proposed model has an improvement in precision, recall, F1 value and accuracy. Compared to the best deep learning method (Bi-LSTM), our proposed model improves 1.45% in precision, 1.39% in recall, 1.42% in F1 value and 1.42% in accuracy.

Table 3 Comparations on dataset 2

Method	*Precise*	*Recall*	*F*1	*Accuracy*
KNN	84.23%	84.09%	84.16%	84.15%
SVM	82.98%	83.12%	83.05%	83.07%
DT	83.76%	83.81%	83.78%	83.79%
Bi-LSTM	87.89%	88.02%	87.95%	87.97%
Transformer encoder	86.45%	86.51%	86.48%	86.49%
TextCNN	85.67%	85.72%	85.70%	85.71%
Proposed model	89.34%	89.41%	89.37%	89.39%

5 DISCUSSION

5.1 Key Findings

Fuzzy neural network is a neural network model based on fuzzy logic, which is capable of handling fuzzy and uncertain information. In false news detection, there are some fuzzy and uncertain features, such as semantic ambiguity and incomplete information. By using Fuzzy neural network, these fuzzy and uncertain features can be better handled to improve the model's ability to recognize false news. And positional self-attentive coding is a variant of self-attentive mechanism which introduces positional information in the attention mechanism. In natural language processing tasks, positional self-attentive coding can help the model better understand the importance and relevance of different positions in the text. For fake news detection, positional self-attention coding can help the model better capture the key information and context in the text, thus improving the model's ability to judge fake news. By combining fuzzy neural networks and positional self-attentive coding, the advantages of fuzzy logic and positional information can be fully utilized to improve the performance of the fake news detection model. This combination can better handle fuzzy and uncertain features while better capturing key information and context in the text, thus improving the accuracy and robustness of fake news detection. The study of fake news detection is critically important and timely due to several societal factors:

Digital Information Explosion: With the rise of the internet and social media, there is an overwhelming amount of information available. This makes it challenging to differentiate between credible news and fake news, leading to a dire need for effective detection methods.

Influence on Public Opinion and Democracy: Fake news can significantly sway public opinion and impact democratic processes, such as elections. The spread of misinformation can manipulate voters' perceptions, posing a threat to the integrity of democratic systems.

Sophisticated Disinformation Campaigns: The increasing sophistication of disinformation campaigns, often powered by advanced technology, threatens to destabilize societies and mislead the public on a large scale.

Erosion of Trust: The prevalence of fake news is contributing to a crisis of trust in media and institutions, highlighting the need for reliable methods to verify information.

Urgency for Real-Time Solutions: The rapid dissemination of fake news through digital platforms necessitates fast detection and response to prevent the widespread acceptance of false narratives.

Ethical and Legal Challenges: The spread of fake news raises ethical questions about the right to accurate information and legal challenges in terms of regulation across different regions.

Given these factors, developing advanced, effective methods for fake news detection is not just a technological challenge but a societal necessity. This research aims to offer a novel solution to support informed and truthful public discourse in our increasingly digital world [20, 21]. Integrating fuzzy neural networks into our fake news detection model indeed increases its complexity, impacting both computation time and the speed of convergence. This complexity stems from the additional fuzzy logic components, which add more parameters and computational layers to the model. Consequently, the model requires more time to process and train, as it deals with the intricacies of fuzzy sets and rules. However, it is important to consider the trade-offs involved. While there is an increase in computational demands, the enhanced ability of the model to process ambiguous and uncertain information - a common characteristic in fake news - can lead to improved accuracy and effectiveness. This benefit is significant for complex tasks like fake news detection. To mitigate the increased computational requirements, we can employ optimization strategies such as refining the fuzzy logic architecture and using advanced training algorithms, ensuring that the model remains efficient while leveraging the advantages of fuzzy neural networks.

5.2 Limitations

The above methods have some limitations in the task of fake news detection.

1) Data requirements: fuzzy neural networks and positional self-attentive coding methods usually require a large amount of labeled data for training. Obtaining large-scale accurate labeled data can be a challenge, especially for tasks such as fake news detection, which requires professionals to accurately categorize the news.

2) Parameter tuning: both fuzzy neural networks and positional self-attention coding methods have some hyper-parameters that need to be tuned, for example, the design of fuzzy sets and fuzzy rules in the fuzzy neural networks, and the attention weights in the positional self-attention coding computation method. Adjusting these hyperparameters may require some experience and domain knowledge.

3) Interpretability: Fuzzy neural networks and positional self-attention coding methods are to some extent black-box

models and it is difficult to explain the decision-making process of the model. This may limit the need for explanatory and interpretable models, especially in some sensitive domains such as news reporting.
4) Computational resources: fuzzy neural networks and positional self-attentive coding methods usually require larger computational resources and time for training and inference. This may limit the application of these methods in resource-constrained environments. In conclusion, combining fuzzy neural networks and positional self-attentive coding for fake news detection is a promising research direction.

5.3 Future Directions

Combining fuzzy neural networks and positional self-attentive coding for fake news detection is a promising research direction. Future directions may include the following:
1) Model optimization: further improve and optimize the structure and parameter settings of fuzzy neural networks and positional self-attention coding in order to improve the performance and generalization ability of the model. Different model architectures, attention mechanisms and fuzzy set designs can be tried to find more effective combinations.
2) Data augmentation and extension: adding more training data, especially labeled data for fake news, to improve the training effectiveness and generalization ability of the model. In addition, data augmentation techniques can be considered to extend the training dataset to increase the robustness of the model.
3) Multimodal fusion: combining text, image, video and other multimodal information for more comprehensive fake news detection. It is possible to explore how fuzzy neural networks and positional self-attentive coding can be applied to multimodal data with effective fusion and representation learning.
4) Explanatory and Interpretable: improve the explanatory and interpretable nature of the model so that it can explain the model's decision-making process and the basis of judgment. This is especially important for the fake news detection task, which can help users understand the judgment results of the model and enhance their trust in the model.
5) Real-time and online learning: for the real-time needs of fake news detection, we can explore how to apply fuzzy neural networks and positional self-attentive coding to online and incremental learning scenarios so that the model can adapt to new fake news forms and changes in a timely manner.

In conclusion, the future development direction is to further improve and optimize the model structure, extend the dataset, multimodal fusion, improve the interpretability and explainability, and explore the aspects of real-time and online learning in order to improve the performance and application of the fake news detection method combining fuzzy neural networks and positional self-attention coding.

6 CONCLUSIONS

The aim of this study is to explore and apply a combination of fuzzy neural networks and positional self-attentive coding to improve the accuracy and robustness of fake news detection. Through experimental evaluation on the fake news detection task, we draw the following conclusions:
First, fuzzy neural networks have good performance in fake news detection. By introducing fuzzy sets and fuzzy rules, fuzzy neural networks can better model and process fuzzy inputs and outputs. The experimental results show that the fuzzy neural network achieves high accuracy and robustness in the fake news detection task.
Second, positional self-attentive coding also plays a key role for fake news detection. Positional self-attentive coding is able to learn the relationships between different positions in the text and calculate the attentional weight of each position based on these relationships. This attentional weight can help the model better understand the semantic and contextual information of the text, thus improving the performance of fake news detection.
Finally, combining fuzzy neural networks and positional self-attention coding can further improve the accuracy and robustness of fake news detection. Fuzzy neural networks are able to capture the relationship between fuzzy inputs and outputs, while positional self-attentive coding is able to capture the relationship between different positions in the text. By combining the two, our model is able to better understand and process fake news text, thus improving the detection performance. In summary, the method proposed in this study combining fuzzy neural networks and positional self-attentive coding achieves good results in the fake news detection task. This approach not only improves the accuracy and robustness of fake news detection, but also provides a valuable reference for further research and application of fake news detection.

In conclusion, this study makes significant contributions to the field of fake news detection. Our approach of combining fuzzy neural networks with positional self-attention mechanisms introduces a new paradigm in processing and analyzing text data for authenticity. This not only advances the theoretical understanding of fake news detection mechanisms but also offers practical solutions with improved accuracy and applicability. We believe that these contributions mark a substantial step forward in the ongoing battle against fake news and misinformation.

7 REFERENCES

[1] Hu, L., Wei, S., Zhao, Z., & Wu, B. (2022). Deep learning for fake news detection: A comprehensive survey. *AI Open, 3*, 133-155. https://doi.org/10.1016/j.aiopen.2022.09.001

[2] Capuano, N., Fenza, G., Loia, V., & Nota, F. D. (2023). Content Based Fake News Detection with machine and deep learning: a systematic review. *Neurocomputing, 530*, 91-103. https://doi.org/10.1016/j.neucom.2023.02.005

[3] Ali, I., Ayub, M. N. B., Shivakumara, P., & Noor, N. F. B. M. (2022). Fake News Detection Techniques on Social Media: A Survey. *Wireless Communications and Mobile Computing,* ID 6072084. https://doi.org/10.1155/2022/6072084

[4] Aïmeur, E., Amri, S., & Brassard, G. (2023). Fake news, disinformation and misinformation in social media: a review. S*ocial Network Analysis and Mining, 13*(1), 30. https://doi.org/10.1007/s13278-023-01028-5

[5] Nurhasanah, N., Sumarly, D. E., Pratama, J., Heng, I. T. K., & Irwansyah, E. (2022). Comparing SVM and Naïve Bayes

Classifier for Fake News Detection. *Engineering, MAthematics and Computer Science (EMACS) Journal, 4*(3), 103-107. https://doi.org/10.21512/emacsjournal.v4i3.8670

[6] Choudhury, D. & Acharjee, T. (2023). A novel approach to fake news detection in social networks using genetic algorithm applying machine learning classifiers. *Multimedia Tools and Applications, 82*(6), 9029-9045. https://doi.org/10.1007/s11042-022-12788-1

[7] Raza, S. & Ding, C. (2022). Fake news detection based on news content and social contexts: a transformer-based approach. *International Journal of Data Science and Analytics, 13*(4), 335-362. https://doi.org/10.1007/s41060-021-00302-z

[8] Chen, M. Y. & Lai, Y. W. (2022). Using fuzzy clustering with deep learning models for detection of COVID-19 disinformation. *Transactions on Asian and Low-Resource Language Information Processing*. https://doi.org/10.1145/3548458

[9] Zhang, Q., Guo, Z., Zhu, Y., Vijayakumar, P., Castiglione, A., & Gupta, B. B. (2023). A deep learning-based fast fake news detection model for cyber-physical social services. *Pattern Recognition Letters, 168*, 31-38. https://doi.org/10.1016/j.patrec.2023.02.026

[10] Xiong, C. J., Goh, K. O. M., & Connie, T. (2022). Deepfakes detection using computer vision and deep learning approaches. *Journal of System and Management Sciences, 12*(5), 21-35. https://doi.org/10.33168/JSMS.2022.0502

[11] Karnyoto, A. S., Sun, C., Liu, B., & Wang, X. (2022). Transfer Learning and GRU-CRF Augmentation for Covid-19 Fake News Detection. *Computer Science and Information Systems, 19*(2), 639-658. https://doi.org/10.2298/csis210501053k

[12] Baydogan, C. & Alatas, B. (2021). Sentiment Analysis in Social Networks Using Social Spider Optimization Algorithm. *Tehnicki vjesnik, 28*(6), 1943-1951. https://doi.org/10.17559/TV-20200614172445

[13] Erdemir, E. & Altun, A. A. (2022). Website Phishing Technique Classification Detection with HSSJAYA Based MLP Training. *Tehnicki vjesnik, 29*(5), 1696-1705. https://doi.org/10.17559/TV-20211227132418

[14] Liang, C., Zhang, Y., Li, X., Zhang, J., & Yu, Y. (2022). FuDFEND: fuzzy-domain for multi-domain fake news detection. *CCF International Conference on Natural Language Processing and Chinese Computing*, 45-57. https://doi.org/10.1007/978-3-031-17189-5_4

[15] Kumar Dixit, D., Bhagat, A., & Dangi, D. (2022). Fake news classification using a fuzzy convolutional recurrent neural network. *Computers, Materials & Continua, 71*(3), 5733-5750. https://doi.org/10.32604/cmc.2022.023628

[16] Wotaifi, T. A. & Dhannoon, B. N. (2022). Improving Prediction of Arabic Fake News Using Fuzzy Logic and Modified Random Forest Model. Karbala International *Journal of Modern Science, 8*(3), 477-485. https://doi.org/10.33640/2405-609X.3241

[17] Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, L., & Polosukhin, I. (2017). Attention is all you need. *Advances in neural information processing systems*, 30.

[18] Teng, L. & Qiao, Y. (2022). BiSeNet-oriented context attention model for image semantic segmentation. *Computer Science and Information Systems, 19*(3), 1409-1426. https://doi.org/10.2298/CSIS220321040T

[19] Hua, X. (2022). DRN-SEAM: A Deep Residual Network Based on Squeeze-and-Excitation Attention Mechanism for Motion Recognition in Education. *Computer Science and Information Systems, 19*(3), 1427-1444. https://doi.org/10.2298/CSIS220322041H

[20] Yuan J., Rahman N., & Mustafa H. (2023). Research on Media Continuance Intention Based on the C-OAR-SE Model. Journal of Logistics. *Informatics and Service Science, 10*(3), 221-234. https://doi.org/10.33168/JLISS.2023.0317

[21] Retkutė K. & Davidavičienė V. (2021). Application of Multichannel Access and Customer Journey Map in the Context of Innovative Business Projects. *Journal of Service, Innovation and Sustainable Development, 2*(1), 100-120. https://doi.org/10.33168/SISD.2021.0109

Contact information:

Ning PAN
Beijing Lurun Guotai Investment Co., Ltd.
E-mail: 536078898@qq.com