FAKE NEWS DETECTION USING NLP

SUBMITTED IN PARTIAL FULFILLMENT OF THE REQUIREMENTS FOR THE

THE AWARD OF

BACHELOR OF ENGINEERING DEGREE IN COMPUTER SCIENCE AND ENGINEERING

BY

EZHILKAVIYA.K

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING
STARLION COLLEGE OF ENGINEERING AND TECHNOLOGY
MANANKORAI,THANJAVUR.

-614206
OCTOBER -2023

Abstract

Fake news is a big problem in every society. Fake news must be detected and its sharing should be stopped before it causes further damage to the country. Spotting fake news is challenging because of its dynamics. In this research, we propose a framework for robust Thai fake news detection. The framework comprises three main modules, including information retrieval, natural language processing, and machine learning. This research has two phases: the data collection phase and the machine learning model building phase. In the data collection phase, we obtained data from Thai online news websites using web-crawler information retrieval, and we analyse the data using natural language processing techniques to extract good features from web data. For comparison, we selected some well-known classification Machine Learning models, including Naïve Bayesian, Logistic Regression, K-Nearest, Multilayer Perceptron, Support Vector Machine, Decision Tree,

Random Forest, Rule-Based Classifier, and Long Short-Term Memory. The comparison study on the test set showed that Long Short-Term Memory was the best model, and we deployed an automatic online fake news detection web application.

Keywords Fake news detection · Information retrieval · Natural language processing · Machine learning

Introduction

The evolution of information and communication technology has dramatically increased the number of Internet users. It transforms the way people consume information and news from traditional to digital, resulting in comfort and speed for both news presenters and newsreaders. In its convenience, the Internet system also generates a lot of fake news content. Fake news has become one of the major concerns as it can destabilize governments that endanger modern society. For example, the electoral campaign in the USA in 2016 had the term “fake news” found to gain much prominence due to the influence of fraudsters. The Internet is a big data source of online news. published on paper. Now newspaper bureaus have moved to online platforms. The readers can easily access from any place at any time via the Internet. People are now comfortable accessing online news and can quickly share the news contents across the social network media such as WWW, Google, YouTube, Google+, Facebook, Twitter, Instagram, and Line. Fake news is a threat to democracy around the world, which has weakened the confidence of governments, newspapers, and civil society. The public’s popularity on social media and social networks has led to the proliferation of fake news with conspiracy theories, distortions, and violent views. Detecting and mitigating the impact of fake news is one of the fundamental problems of modern times and is gaining widespread attention. While fact-checking websites such as big companies like Google, Facebook, and Twitter, have taken some preliminary steps in dealing with fake news. Many communities include machine learning, databases, journalism, political science, and many others, pay attention to aspects of fake news as an interdisciplinary topic. There is still a lot to do to cope with the fake news issues .Many researchers have proposed various machine learning approaches for fake news detection. Shu et al. proposed a fake news detection framework exploiting social context called a tri-relationship embedding framework. The model was based on publisher–news relations and user–news interactions simultaneously for fake news classification. They demonstrated that the proposed method significantly outperforms other existing fake news detection approaches proposed a neural network-based model for fake news detection with generated comments for news articles to help classification. A fake news stance detection using deep learning architecture based on convolutional neural networks and long short-term memory (CNN-LSTM).

The method in passed the non-reduced feature set with and without pre processing to the neural network. The research used the principal component analysis (PCA) for dimensionality reduction, which increases the classifier performance because it removes the irrelevant, noisy, and redundant features from the feature vector. Akhter et al. proposed an annotated corpus of Urdu news articles for the fake news detection tasks. The researchers used ensemble learning methods based on Naïve Ba ye, Decision Tree, and Support Vector Machine to improve the fake news detection system performance.

Pre-trained Transformer (GPT). The researchers used COVID-19 news open datasets translated to Thai and pre training Thai COVID-19 deep learning models. To fine-tune for a local dataset, the researchers used additional data by crawling Thai texts from social media them as fake and real samples. The best results from their experiments achieved the best accuracy performance of 72.93%. There was no report of real use cases for Thai fake news in the research.

Building fake news detection is a challenging task. It will be even more difficult for Thai fake news detection in the real situation, as the Thai language is one of the most complex languages with no space between words. In this research, we propose a framework to create an automatic online Thai fake news detection system. The proposed framework comprises three modules: information retrieval, natural language processing, and machine learning. Construction of the online fake news has three phases: data collection, data preparation, and machine learning model. The contributions of this research are as follows: We propose a framework of online fake news detection as the main contribution. In this research, a feature selection algorithm is also a result of natural language analysis. To build Thai fake news detection, we collected a dataset them as fake news.

Literature Reviews

Fake News

In the digital age, more and more people use their daily lives to connect to the Internet and social networks. People are using the Internet on the rise with the convenience of delivering, accessing, and sharing news via the Internet and social networks, which makes it easy to spread information without any restrictions while posting it on these platforms. However, the information that is published may contain both real news and fake news. Some malicious users take advantage of these platforms by generating fake news, spreading them on the Internet and social media networks to damage the reputation of individuals, businesses, and politics.

Misinformation can appear in different formats and domains, such as fake news, click baits, and false, and much of the previous research has focused on model specific to a single domain . These domains may have different formats, such as long articles versus short headlines and tweets, and their exact purpose like “This is a fake” vs. “Click Bait”; however, they have the same goal of deceiving the readers. As a result, content that exhibits similar linguistic features, such as the use of exciting themes to arouse curiosity or intense emotional responses from readers . Therefore, many researchers proposed a way to detect fake news to stop the distribution of fake news. Online news is dynamics during propagation on social media. Malicious users can diverge from the original and create fake news. It makes detecting fake news automatically from the Internet a challenging task in detecting fraud.

Creating automatic fake news or misinformation detection involves many theories and practices. The main disciplines may include information retrieval, natural language processing, and machine learning.

Information Retrieval

Databases store the information in a structured manner in many documents. When searching documents, it is a problem to find information needed, such as search terms or sample documents. An information retrieval system (IR) is a software system that provides an access to documents to manage and store them. An IR system is a branch developed in conjunction with a database system. IR can be considered as the science of searching for information in documents, manual document searching, and descriptive metadata search, and for databases of text, images, or sound . It is the activity of obtaining information from information system resources relevant to the information needed. A query can be full-text or other content indexing. An automated IR system can reduce data overload. For basic concepts in IR, documents can be explained by a set of terms representing a document is called index terms. Different index terms are relevant when used to describe the content of a document. This effect is assigned a numerical weight to each document index, such as term frequency and inverse document frequency (TF $^{\times}$ IDF).

IR models have three types: Boolean Model, Vector Model, Probability Model. The Boolean model is an exact match between the index terminology and the search terms. Boolean information retrieval predicts each document whether it is relevant or not relevant to the document query. For a vector information retrieval model, vocabulary, or word (term) is used instead of attributes. The searched document comprises words converted to numbers called term frequency or weight values. The weight values are a substitute for document queries. With the weight values, distance formula or similarity measure calculates the relationship between the query against the document in the database. The vector information retrieval emphasizes the frequency of the words contained in the document and the effect on the weighting of the term against the word count of the document word weight. The third model, the probability information retrieval model is based on a user query. The probability information retrieval model sequences the documents according to the probability based on their relationship or relevance to the query text, where high probability means high relevance. The accepted probability calculation method is calculated from the word frequency data.

Natural Language Processing

Natural Language Processing (NLP) is a sub-branch of linguistics, computer science, data engineering, and artificial intelligence. NLP relates to the interaction between humans and computers. NLP is a method for processing and analyse large amounts of natural language data. NLP has many applications such as machine translation, speech recognition, sentiment analysis, automatic question and answer generation, automatic message digest, chat bot, intelligence, text classification.

In the NLP, one crucial step is text extraction, a pre processing step for using the analysis of text, documents, news, and information before implementing the clustering, classification, or other machine learning tasks. The fundamental pre processing step for NLP includes word segmentation, tokenization, word stopping, word stemming, term frequency weighting, term frequency, and inverse document frequency weighting. Some advanced NLP techniques may include more complex tasks in the pipeline, such as parts of speech tagging, dependency parsing, named entity recognition, and conference resolution. Advanced NLP techniques may employ lexical analysis, syntactic analysis, semantic analysis, disclosure integration, and pragmatic analysis.

The Thai language is a language that has words in continuous place without space in consecutive sentences in the documents. For analysis, the Thai documents need to break down into a single word like English. Word segmentation is the separation of each word from sentences, which still has correct meaning by using a dictionary database of words. There are many techniques for word segmentation including Longest Word Pattern Matching, Shortest Word Pattern Matching, Word Wrapping, Probabilistic Word Segmentation, Back Tracking, Feature-based, and Machine Learning based techniques.

Machine Learning

Machine learning is a study field in computer science, which involves creating adaptive programs that can learn via training data. There are many forms of machine learning, including supervised learning, unsupervised learning, semi supervised learning, and reinforcement learning. Normally, building a machine learning model starts with data preparation for two sets: training data and test data. Machine learning learns from training data. The users evaluate the trained machine learning model using the test data. The evaluation by the test data is to make sure that we can use the trained model to predict the future unseen data with confidence. Training the machine learning model is a search for optimal parameters. The users seek the most suitable machine learning model parameters. The users choose a machine learning model for a proper task as different techniques will suit different tasks.

In this paper, we focus on supervised learning that include Logistic Regression (LR), K-Nearest (KNN), Rule-Based Classifier (RBC), Decision Tree (DT), Random

Forest (RF), Naïve Bayesian (NB), Multilayer Perceptron (MLP), Support Vector Machine (SVM), and Long Short Term Memory (LSTM).

Logistic Regression (LR) is a mathematical modelling based machine learning used to describe the relationship of several independent variables to a dependent dichotomous (binary) variable. We can train LR as classifier as it is a suitable regression analysis for the dependent dichotomous variable. Logistic regression can describe the relationship between one or more dependent binary variables and independent variables that specify at least one sequence, range, or ratio level.

KNN is a simple non-parameter classification method. The KNN is a case-based learning method that maintains all training data for the classification task. To use KNN, we need to choose a suitable *K* value, and the classification result depends on this value. There are many ways to select a *K* value, but the easy way is to run the algorithm multiple times with different *K* values and choose the most effective one. To classify data using a KNN classifier, we need three things: stored training data, *K* value, distance, or similarity metric. The KNN performs as follows. read in a data record to classify .compute the distance between the classifying data record to all stored training data .select the *K* smallest distance. And classify the classifying data based on the majority vote from the *K* nearest data records' labels.

Rule-Based Classifiers (RBC) comprise a rule set in the form IF *X*, then *Y*. Using classification training dataset, we can train a rule-based system to become an RBC. RBC needs a rule-based algorithm to generate a rule set as a classification scheme defined as a set of IF-THEN rules. We then can use the ruleset to classify each instance in the dataset. CN2 is one of the most widely used as a rule induction algorithm. CN2 Rule Induction is a rule-based algorithm of the rule based classifiers. CN2 uses the heuristic function, such as Entropy, Laplace, and Accuracy, to terminate the search during rule formation based on the noise approximation present in the data. The specified rules may not correctly classify all training samples. However, it works fine with the new unseen data. The CN2 accepts only rules of exceptional precision so it can deal with noise. Besides, CN2 can create a sorted or unordered rule list.

Decision Tree Classifier (DTC) is a tree-like model represented as a recursive partition of the data space. A decision tree consists of a most discriminant node that forms a rooted tree. At the top, a tree starts with a root node that does not have an incoming branch or link, or edge. All other nodes have only one incoming edge. Nodes with outgoing edges are called intermediate or test nodes. At the lowest levels, nodes are called leaves, which are decision nodes. There are many decision tree induction algorithms; some famous algorithms are ID3, C4.5, and C5.

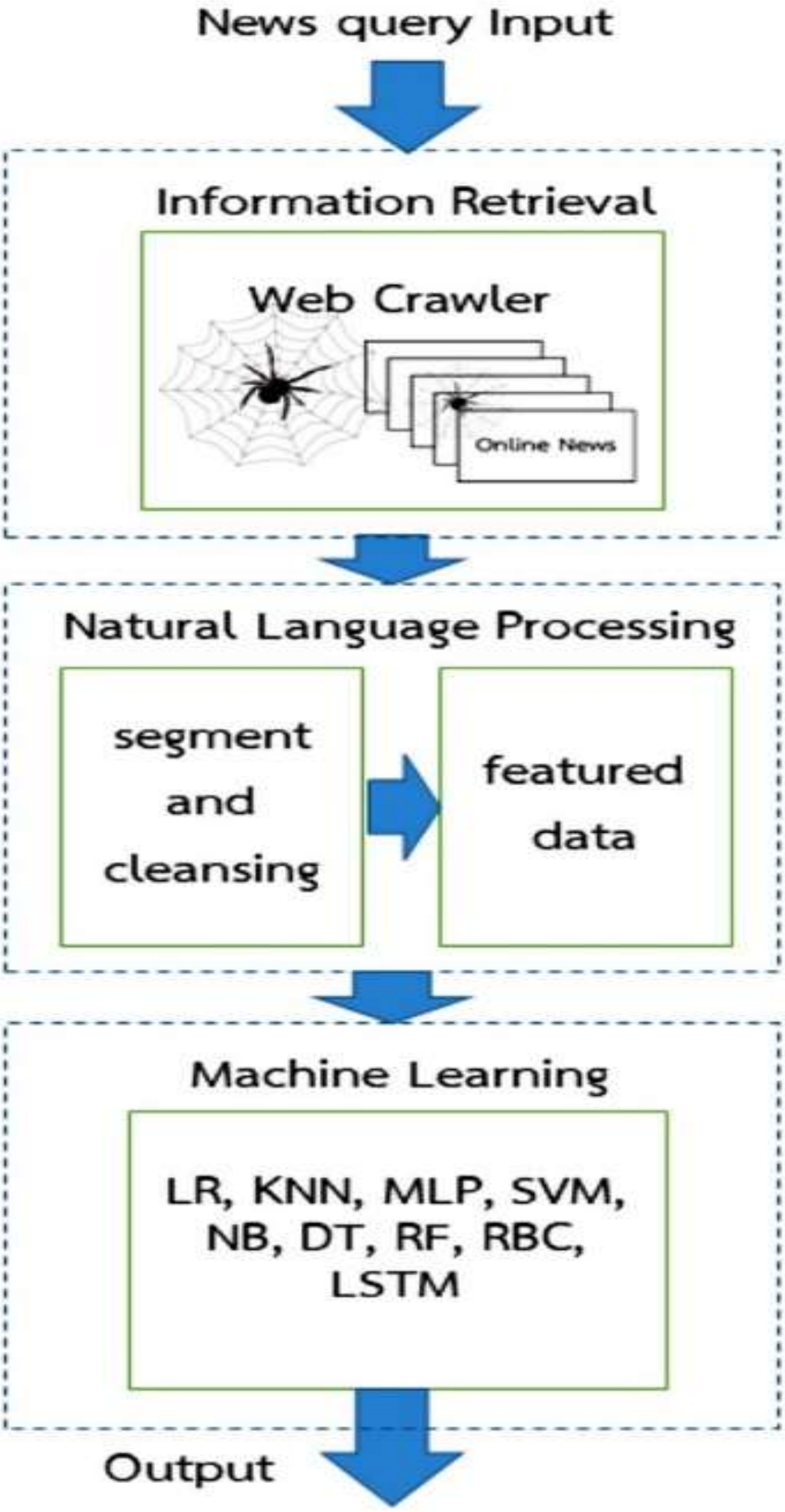
Random Forest (RF) is one of the best algorithms for classification tasks. The basic idea behind RF is that a group of weak learners can form a strong learner. RF can classify large datasets with high accuracy and precision. RF acts as a classifier with every tree dependent on a random vector value. RF generates many decision trees at the time of training, and the outcome of the modalities predicted by each tree created using bootstrap samples of training data and random selection of attributes in tree induction. Prediction is formed by combining using majority vote or averaging all decision trees.

Naïve Bayesian (NB) is an easy learning probabilistic based algorithm that uses Bayes' rule in conjunction with the explicit assumption that attributes are conditionally independent of each other. Based on training data, NB estimates the posterior probability of each class, *y*, of a given object, *x*. We can use the estimation for classification applications. Because of its computational efficiency and many other desirable properties, NB appears as an acceptable solution in many practical implementations.

Multilayer Perceptron (MLP) is one type of artificial neural network (ANN) based on simulating the function of the human brain using a computer program. The goal of ANN is to make computers as intelligent as humans are. ANN can learn from training data and recall back knowledge to apply to the specific trained problems such as Classification, Regression, and Clustering. MLP is often referred to as a "black box" because of its functionality. MLP, sometimes called a feedforward network, has several calculation steps. It starts with the input data entered at the input layer having synaptic weight linked to neurons in the hidden layers. MLP may have several hidden layers depending on the complexity of data. Each hidden layer has synaptic weights connected to the next layer. The outputs from the previous layer act as the input to the next layer. The signal reaches the output layer, where the prediction output goes out from the neural network.

Support Vector Machine (SVM) is based on the learning of statistical theory. Several researchers applied SVM to many applications in data classification or pattern recognition. SVM theoretical concepts are as follows. Structural Risk Minimization is a concept that expresses the extent of the risk or the likelihood of learning errors. The SVM learning process determines the function of decision-making to minimize the error rate. The kernel function is an important concept that supports a vector machine technique. A kernel function maps data from input space to feature space to create non-linear decision-making functions to data in the leading space. Optimal margin hyperplane is a crucial concept of vector machine support techniques. The learning process of SVM is to find the plane with the maximum margin, in which it can separate the data into two groups apart and solve the problem of overfitting .

Long Short-Term Memory (LSTM) is a deep learning model in a recurrent Neural Network (RNN) group. RNN provides hidden state feedback as input that makes it possible to capture the dependency of sequence data such as time series and natural languages. RNN is not only to process a single data point but also to process sequential data. LSTM, developed to solve the problem of exploding and vanishing the slope error faced in traditional RNN, is well suited for classification, processing, and prediction based on time series data as there may be an unknown period between events in time series. One can use LSTM for many tasks such as sentiment analysis from documents, handwriting recognition, speech recognition, and anomaly detection of network data.


News query Input
Information Retrieval
Web Crawler
Online News
Natural Language Processing
segment and cleansing
featured data
Machine Learning
LR, KNN, MLP, SVM, NB, DT, RF, RBC, LSTM
Output

It is a challenging task for Thai language processing because the Thai language has no space between words. We applied as a tool for Thai word segmentation. We used the maximum matching method for Thai word segmentation and a custom dictionary with the size of vocabularies of 75,936 words used in this study. Figure shows the flowchart of the natural language processing framework.

Table 1 Confusion matrix

	Predicted classes				
Actual classes	C_1	C_2	C_3	…	C_G
C_1	$T_{1,1}$	$F_{1,2}$	$F_{1,3}$	…	$F_{1,G}$
C_2	$F_{2,1}$	$T_{2,2}$	$F_{2,3}$	…	$F_{2,G}$
C_3	$F_{3,1}$	$F_{3,2}$	$T_{3,3}$	…	$F_{3,G}$
…	…	…	…	…	.
C_G	$F_{G,1}$	$F_{G,2}$	$F_{G,3}$	…	$T_{G,G}$

For feature extraction, Algorithm 2, we use words that usually appear on fake or real news. The feature extraction takes acts when the web crawlers retrieve the news contents relevant to the query based on the truncated cosine similarity defined in (3). We used (7)–(13) for feature extraction from each news document. The data extracted are expected to have discriminatory characteristics of fake and real news. Please be informed that the context of fake and real news may be different in each country. The predefined negative words and positive words may vary in other countries. Here the positive and negative words are for Thai news.

The sample negative words, translated from Thai, are as follows: [ambiguous facts, ancient stories, artificial news, bad information, bad news, baseless, brag, but did you know, cannot cure disease, cannot do it, casual, catch pontoon, claims, cut paste, deceitful, deception, defamation, distorted messages, do not believe, do not share, does not exist, don’t become victims, editing, fake, fake events, fake information, fake messages, fake news, fake news messages, fake news stories, fake stories, false, false beliefs, false facts, false information, false news, false reports, false statement, false stories, falsely, fraud, fraudulent web, garbage, incorrect facts, insecure facts, insecure information, insecure news, insufficient data, invalid information, is not true, lie, madden news, make a story, misinformation, misleading information, misrepresentation, mistakes of information, misunderstanding, negative news, no indication, no information, not qualified, not real, not real information, not real news, not true, not trustworthy, prank, propaganda, scam, slang, slogans, suspicious information, uncertain facts, uncertain information, uncertain news, unclear information, uncoordinated data, unreliable facts, unreliable information, unreliable news, untrue facts, valuable information, worthless facts, worthless news, wrong news, wrong ways].

The sample positive words, translated from Thai, are shown as the following list: [authenticity, confirmed to be true news, no distortion, no fake, no false, no false news, non fake news, non-fake news, non-false news, not fake news, not false, real data, real information, real message, real news, shareable, true, true message, true news, true story, verified news].

Using Algorithm 1, Algorithm 2, and Algorithm 3, we collected data for 41,448 samples with three groups: real, fake, and suspicious. Each group has 13,816 records equally distributed. We separated data into three sets: training set, validation set, and test set. The number of training data was 20,310. The number of validation data was 8,704, and the number of test data was 12,435. Table 2 shows sample data collected to build machine learning for fake news detection. Table 3 shows training, validation, and test sets.

Table 2 Sample data collected

Labels	No. samples
Fake	13,816
Real	13,816
Suspicious	13,816
Total	41,448

Table 3 Training, validation, and test sets

Data sets	No. samples	Ratio
Training	20,723	0.50
Validation	8290	0.20
Test	12,435	0.30
Total	41,448	1.00

Extracted Feature Data

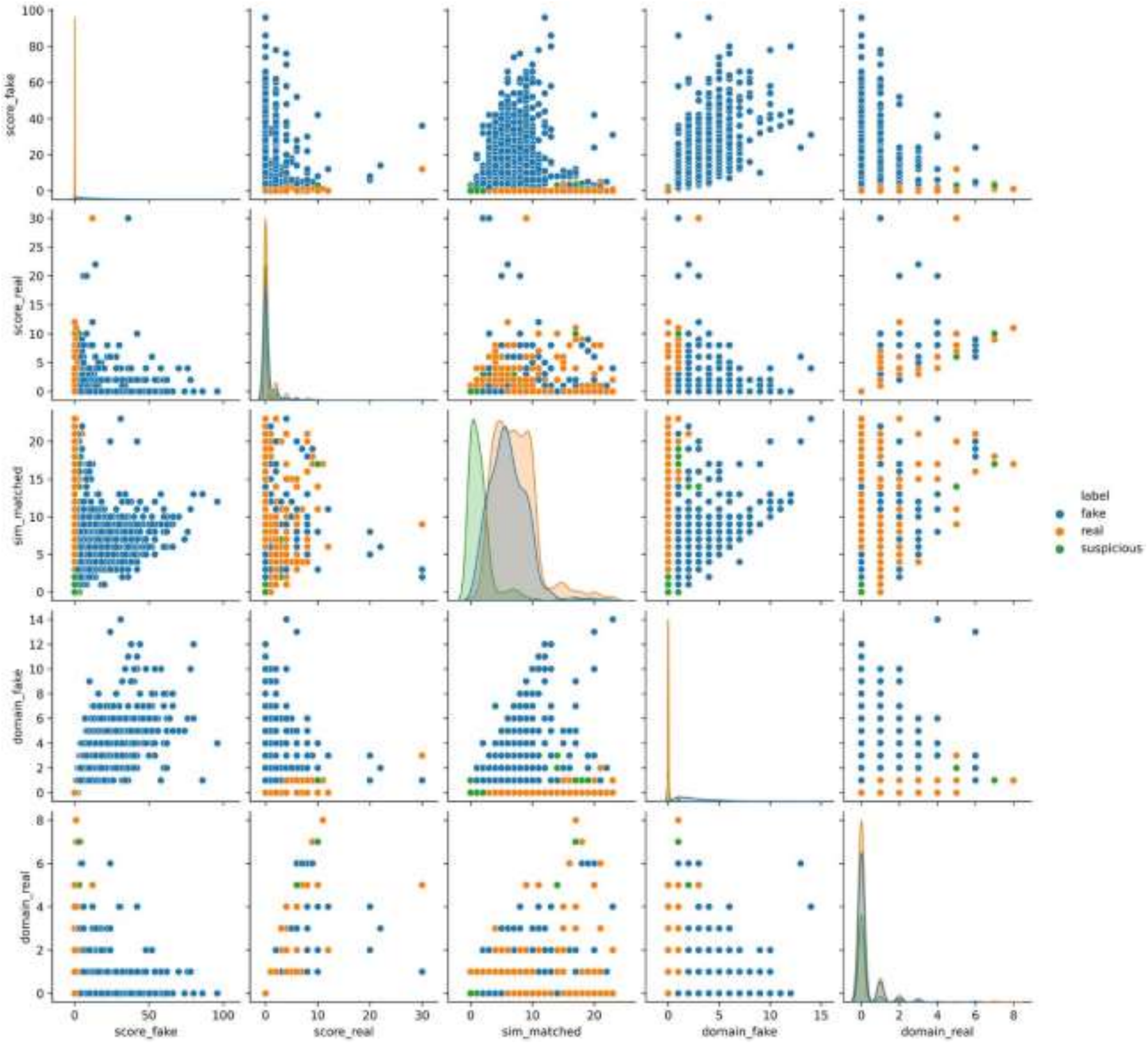
NLP analyse the retrieved data from web crawling. Word segmentation separates text into word tokens. The cleansing process further cleans segmented tokens by removing unnecessary words and characters. The feature extraction process extracts import characteristics from the news content. The extracted features in this research comprise five characteristics: score fake, score real, sim matched, domain fake, and domain real. score fake represents the count of negative words and fake group words that appear on the retrieved news contents. Also, score real is the count of positive or authentic group words found on the retrieved news contents. The sim matched feature is the accumulative cosine similarity between news query and the retrieved news contents. Besides, domain fake and domain real represent the number of websites or the length of domain websites that have fake news and real news, respectively

The featured data include score fake, score real, sim matched, domain fake, and domain real. The targeted classes comprise fake, real, and suspicious. It is worth noting that the extracted features correlate with the targets. The sim matched shows a positive correlation to fake class as well as real class. score fake and domain fake features have 0.7 and 0.76 having predictive influence with class fake. Besides, score real and domain real features have a positive correlation of 0.16 and 0.11 with the class real. It implies that the data can represent fake and real classes quite well. However, class suspicious has a negative correlation with features. It would be difficult to differentiate the suspicious group.

Figure 5 shows the scatter joint plot of clustered feature data. The data clustering shows that it is possible to build a classifier to differentiate the three classes. Class fake

Table 4 Feature data correlation

	Score fake	Score real	Sim matched	Domain fake	Domain real	Fake	Real	Suspicious
Score fake	1.00	0.07	0.35	0.91	0.15	0.70	− 0.43	− 0.35
Score real	0.07	1.00	0.16	0.10	0.86	0.04	0.16	− 0.22
Sim matched	0.35	0.16	1.00	0.37	0.22	0.27	0.30	− 0.65
Domain fake	0.91	0.10	0.37	1.00	0.20	0.76	− 0.47	− 0.38
Domain real	0.15	0.86	0.22	0.20	1.00	0.11	0.11	− 0.25
Fake	0.70	0.04	0.27	0.76	0.11	1.00	− 0.62	− 0.49
Real	− 0.43	0.16	0.30	− 0.47	0.11	− 0.62	1.00	− 0.38
Suspicious	− 0.35	− 0.22	− 0.65	− 0.38	− 0.25	− 0.49	− 0.38	1.00



seems to be well separate from the others, while real and suspicious seem to have an overlapped characteristic.

Pre processing Setting for Machine Learning Models

In this research, we performed experiments based on two groups of machine learning. The first group was traditional machine learning comprising LR, KNN, NB, MLP, RF, and RBC. The second group was the deep learning LSTM model recurrent-based model. For the first group, the input to the models was the featured data extracted from the NLP module. Unlike the traditional models, the input of the LSTM model was a sequence of text content concatenated from retrieved news descriptions. LSTM model used the relevant news content and classified it into fake, real, or suspicious.

There were 952,387 total trainable parameters for the LSTM model. The details of LSTM model settings were as shown in Table 5.

Learning Model Comparisons

After data collection, feature extraction, and data analysis, we built a fake news detection system. We performed model comparisons to choose the best model as a classifier in our news detection system. We selected open-source tools to construct a fake news detection system. The data analysis and machine learning model tools include LR, MLP, SVM, DTC, RF, NB, KNN, RB, and LSTM. The performance metrics used include accuracy, precision, recall, and

	Precision	Recall	*F*-measure
Fake	0.99	0.99	0.99
Real	0.89	0.99	0.94
Suspicious	0.99	0.90	0.94
Weight	0.96	0.96	0.96
Accuracy		0.96	

Table 8 Test performance of RF

	Precision	Recall	*F*-measure
Fake	0.99	1.00	1.00
Real	0.90	0.99	0.94
Suspicious	1.00	0.91	0.95
Weight	0.97	0.96	0.96
Accuracy		0.96	

Table 9 Test performance LSTM

	Precision	Recall	*F*-measure
Fake	1.00	1.00	1.00
Real	1.00	1.00	1.00
Suspicious	1.00	1.00	1.00
Weighted	1.00	1.00	1.00
Accuracy		1.00	

*F*1-measure. Tables illustrate sample of the test performance results based on RBC, SVM, RF, and LSTM, respectively. It is noticed that the sampled models can classify fake and suspicious classes with high precision and F-measure, while they achieve lower scores with the real class. Table shows the summary results of all machine learning models. Figure shows a box plot based on test accuracy for 10-fold-cross-validation. It confirms that LSTM was the best model for achieving a perfect accuracy score. It can be seen that a deep learning LSTM model yields the highest accuracy, precision, recall, and F-measure with a perfect score; all accuracy, precision, recall, and f-measure are 1.00. MLP, RF, and SVM are among the second group with accuracy, precision, recall, and f-measure of 0.96-0.97. NB has the least accuracy, precision, recall, and f-measure, 0.78, 0.85, 0.78, and 0.79, respectively.

Discussion

Fake news data are very dynamics. It is not an easy task to build a fake news detection system that generalizes all unseen data. Our idea is to exploit the news data on the Internet and social media by using it as inputs fed to the

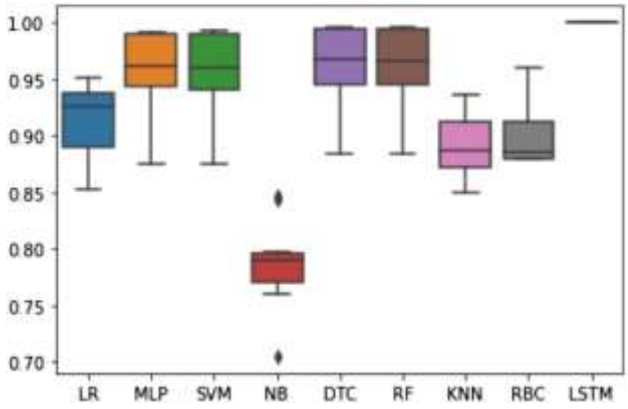


Table 10 Machine learning model comparisons

Metrics |Models	NB	LR	MLP	SVM	DT	RF	KNN	RBC	LSTM
Accuracy	0.78	0.92	0.96	0.96	0.92	0.96	0.93	0.90	1.00
Precision	0.85	0.92	0.96	0.96	0.92	0.97	0.93	0.92	1.00
Recall	0.78	0.92	0.96	0.96	0.92	0.96	0.93	0.90	1.00
F-measure	0.79	0.92	0.96	0.96	0.92	0.96	0.93	0.91	1.00

classifier. As discussed in the data preparation process, we used web crawler-based information retrieval to retrieve the data that contain fake and real news. The feature extraction step extracts news data for five features based on fake news score, real news score, similarity matching, length of fake news domain, and length of real news domain. The extracted feature data have highly distinguished characteristics, as shown in the subsequence machine learning that can perform a classification task with higher accuracy than 90% for most of the classifier models except for the NB. It confirms that the proposed NLP-based feature selection is suitable for the fake news classification task. Besides, LSTM with concatenated text from relevant news having high similarity to the news query achieved best with a perfect test score for all metrics, including accuracy, precision, recall, and *F*-measure.

It is worth noting that the rule-based classifier provides a good feature as an explainable fake news detector. If–Then rules are good for reasoning why the classifier has such an answer to the query. A sample of If–Then rules extracted from the data are listed below.

IF score fake ≥ 9.0 AND score real ≤ 2.0 THEN label=fake

IF score fake ≥ 7.0 AND domain fake ≤ 106.0 THEN label=fake

IF score fake ≥ 3.0 AND score real ≤ 4.0 THEN label=fake

IF score fake ≥ 4.0 AND sim matched ≤ 5.0 THEN label=fake

IF score fake ≥ 4.0 AND score real ≤ 6.0 THEN label=fake

IF domain fake ≥ 29.0 AND domain real ≤ 22.0 THEN label=fake

IF score fake ≥ 4.0 AND sim matched ≥ 8.0 THEN label=fake

IF sim matched ≥ 10.0 AND domain fake ≤ 13.0 THEN label=real

IF score fake ≤ 1.0 AND score real ≥ 2.0 THEN label=real

IF score real ≥ 8.0 AND score fake ≤ 7.0 THEN label=real

IF score fake ≤ 2.0 AND score real ≥ 3.0 THEN label=real

IF score fake ≤ 1.0 AND sim matched ≥ 3.0 THEN label=real

IF domain real ≥ 20.0 AND score real ≤ 2.0 THEN label=real

IF sim matched ≤ 14.0 AND sim matched ≥ 2.0 THEN label=real

IF score fake ≤ 1.0 AND sim matched ≤ 2.0 THEN label=suspicious

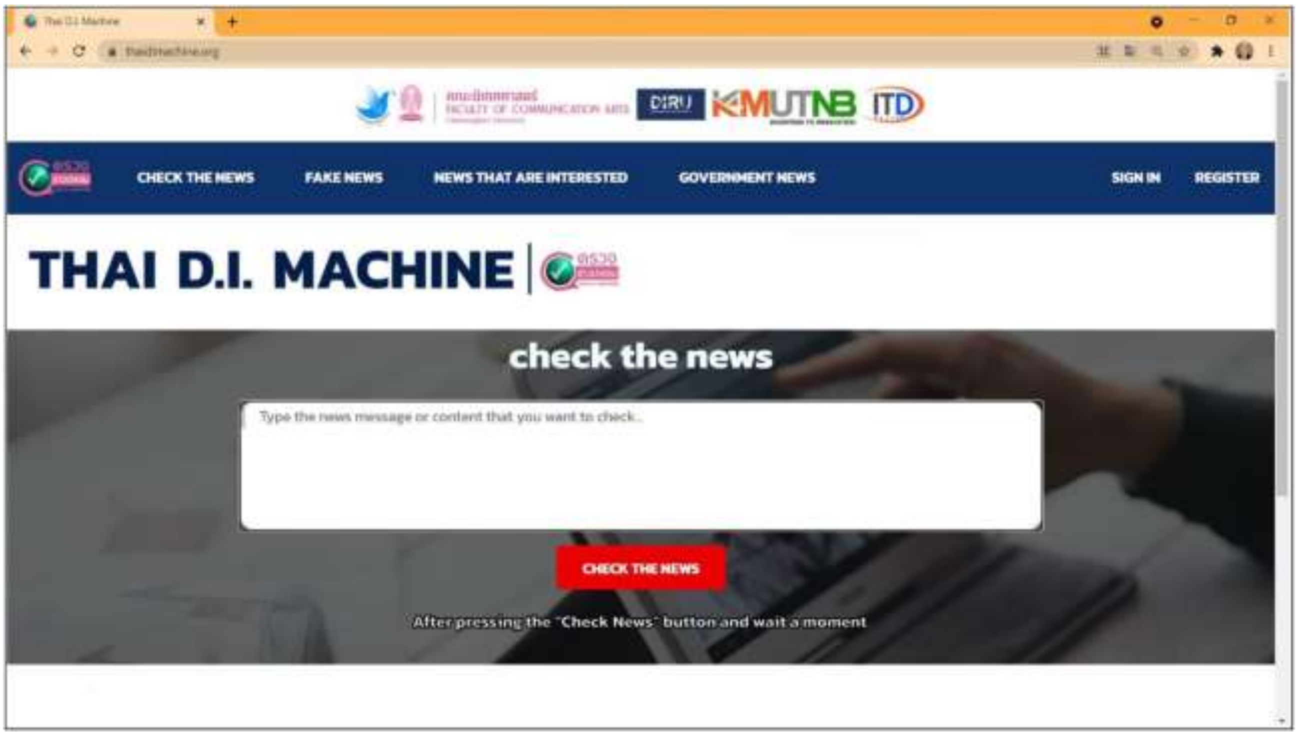
IF sim matched ≤ 4.0 AND domain fake ≤ 13.0 THEN label=suspicious

IF score fake ≤ 1.0 AND domain fake ≥ 15.0 THEN label=suspicious

IF score fake ≤ 2.0 AND score fake ≥ 2.0 THEN label=suspicious

IF domain real ≥ 26.0 AND score fake ≤ 4.0 THEN label=suspicious

IF domain fake ≤ 13.0 AND sim matched ≥ 6.0 THEN label=suspicious



Type the news message or content that you want to check..

After pressing the "Check News" button and wait a moment

IF TRUE THEN label = suspicious

The above rule set is an ordered rules list in which the last rule is the default rule. To decide for each input featured datum, the system checks which rule matches or covers the input datum. If the input datum matches the condition of a rule (TRUE statement), the decision is the label output from the conclusion part of the covered rule. If the datum matches a rule, the decision is made based on the covered rule. There is no other rule needed to check. However, if no rule covered the input datum, the default rule “IF TRUE THEN label = suspicious” will activate. The decision label will be “suspicious.”

Fig. 7 The automatic online Thai fake news detection

Web Application

After data pre processing and machine learning model phases, we designed and developed a fake news detection system using the best-trained machine learning model. For stability, we designed cloud-based online fake news detection. We used the following tools for system development: Ubuntu 20.04 operating system, MongoDB database system for storing news data, Python for information retrieval, natural language processing, and machine learning, Django web framework for frontend web development, and Apache2 as a web server.

The main functions of the system include the query users entered for checking fake or real news. The web application will take the user query to analyse and return the response result with related news websites sorted based on the similarity. The user enters a news query via the text area input form, then the system in parallel sends out web-crawler information retrieval agents to fetch related news from the web and social media. The returned relevant news list is processed via the NLP module to get featured data and fed to the machine learning prediction module. The whole process time may take about 3–10 s to respond, depending on how popular the news query.

It is noticed that we used LSTM instead of BERT and GPT because we use machine learning for classifying the type of news. When a user enters a news query into the system, the user expects a fast response as the best user experience. Having too many parameters, BERT or GPT may not respond quickly enough for classifying news. Just do a classifying job, then we chose LSTM instead.

The web application provides known fake and real news articles, which are currently in the attention of social media communities.

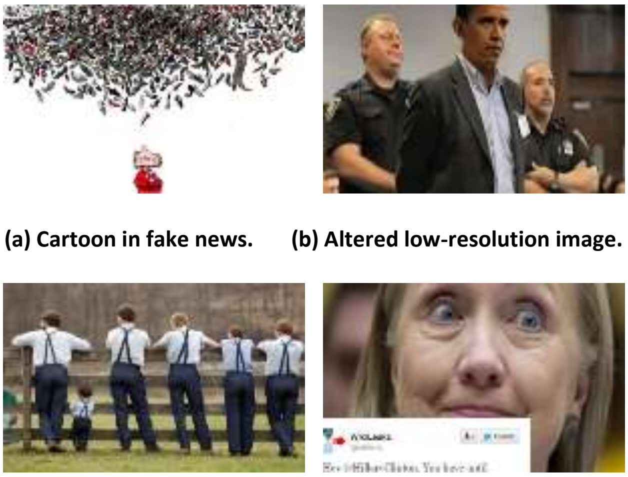
Conclusion

Detecting fake news is a difficult task as the news stories are very dynamic. This research proposes a new robust method to tackle fake news or misinformation. We employ three main techniques to build automatic online fake news detection. In our methodology, first, we use Information Retrieval as a mechanism to retrieve data from an online news website and social media. Next, the natural language processing analyse the retrieved news, which results in feature data that are well distinguished. Lastly, machine learning receives the feature data and classifies the news articles into three classes: real, fake, and suspicious. We used a web robot to crawl data for 41,448 samples and pre-classified them into real, fake, and suspicious classes. The number of data samples in each group is balanced. We

separate the data into three sets: training set, validation set, and test set, each for 50%, 20%, and 30%, respectively. The machine learning models used in the study were Logistic Regression (LR), KNN, Naïve Bayesian (NB), Multilayer Perceptron (MLP), Random Forest (RF), Rule-Based Classifier (RBC), and Long Short-Term Memory (LSTM). We found that LSTM was the best model that achieved 100% on test data measured by accuracy, precision, recall, and f-measure this research.

Convolutional Neural Networks for Fake News Detection:

Fake news identification from online social media is extremely challenging due to various reasons. Firstly, it’s difficult to collect the fake news data, and it is also hard to label fake news manually [43]. News that appears on Facebook and Twitter news feeds belongs to private data. To this context so far, few large-scale fake news detection public dataset really exists. Some news datasets available online involve a small number of the instances only, which are not sufficient to train a generalized model for application. Secondly, fake news is written by human. Most liars tend to use their language strategically to avoid being caught. In spite of the attempt to control what they are saying, language “leakage” occurs with certain verbal aspects that are hard to monitor such as frequencies and patterns of pronoun, conjunction, and negative emotion word usage [10]. Thirdly, the limited data representation of texts is a bottleneck of fake news identification. In the bag-of words approach, individual words or “n-grams” (multiword) frequencies are aggregated and analyse to reveal cues of deception. Further tagging of words into respective lexical cues for example, parts of speech or “shallow syntax” [28], affective dimensions [42], and location-based words [32] can all provide frequency sets to reveal linguistic cues of deception [31], [14]. The simplicity of this representation also leads to its biggest shortcoming. In addition to relying exclusively on language, the method relies on isolated n-grams, often divorced from useful context information. Word embedding techniques provide a useful way to represent the meaning of the word. In some circumstances, sentences of different lengths can be represented as a tensor with different dimensions. Traditional models cannot handle the sparse and high order features very well.



(a) Cartoon in fake news. (b) Altered low-resolution image.

(c) Irrelevant image in fake news. (d) Low-resolution image.

Though the deceivers make great efforts in polishing fake news to avoid being found, there are some leakages according to our analysis from the text and image aspect respectively. For instance, the lexical diversity and cognition of the deceivers are totally different from the truth teller. Beyond the text information, images in fake news are also different from that in real news. As shown in Fig. I, cartoons, irrelevant images (mismatch of text and image, no face in political news) and altered low-resolution images are frequently observed in fake news. In this paper, we propose a TI-CNN model to consider both text and image information in fake news detection. Beyond the explicit features extracted from the data, as the development of the representative learning, convolutional neural networks are employed to learn the latent features which cannot be captured by the explicit features. Finally, we utilize TI-CNN to combine the explicit and latent features of text and image information into a unified feature space, and then use the learned features to identify the fake news. Hence, the contributions of this paper are summarized as follows:

We collect a high quality dataset and take in-depth analysis on the text from multiple perspectives.

Image information is proved to be effective features in identifying the fake news.

A unified model is proposed to analyse the text and image information using the convolutional neural networks.

The model proposed in this paper is an effective way to recognize fake news from lots of online information.

In the rest of the paper, we first define the problem of fake news identification. Then we introduce the analysis on the fake news data. A unified model is proposed to illustrate how to model the explicit and latent features of text and image information. The details of experiment setup is demonstrated in the experiment part. At last, we compare our model with several popular methods to show the effectiveness of our model.

II. RELATED WORK

Deception detection is a hot topic in the past few years. Deception information includes scientific fraud, fake news, false tweets etc. Fake news detection is a subtopic in this area. Researchers solve the deception detection problem from two aspects: 1) linguistic approach. 2) network approach.

A. Linguistic approaches

Bing Liu et.al. analyse fake reviews on Amazon these years based on the sentiment analysis, lexical, content similarity, style similarity and semantic inconsistency to identify the fake reviews. Hai et al. [13] proposed semi-supervised learning method to detect deceptive text on crowdsourced datasets in 2016.

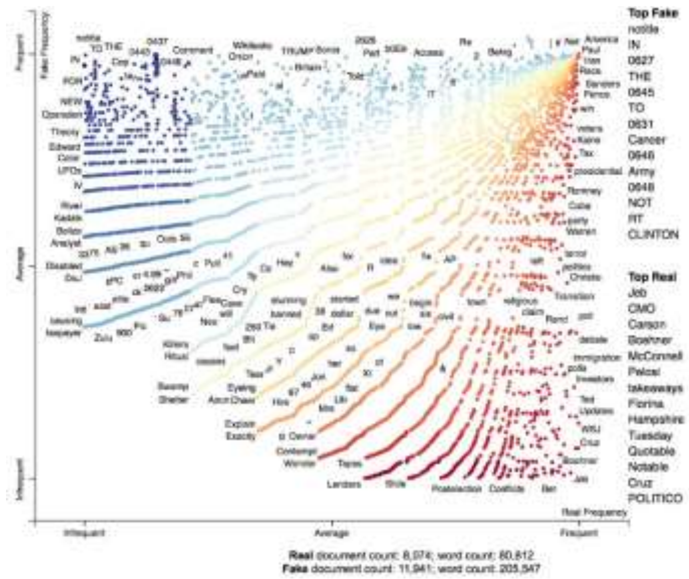
The methods based on word analysis is not enough to identify deception. Many researchers focus on some deeper language structures, such as the syntax tree. In this case, the sentences are represented as a parse tree to describe syntax structure, for example noun and verb phrases, which are in turn rewritten by their syntactic constituent parts [9].

B. Network-based approaches

Another way to identify the deception is to analyse the network structure and behaviours, which are important complementary features. As the development of knowledge graph, it will be very helpful to check fact based on the relationship among entities The methods based on the knowledge graph analysis can achieve 61% to 95% accuracy. Another promising research direction is exploiting the social network behaviour to identify the deception.

C. Neural Network based approaches

Deep learning models are widely used in both academic community and industry. In computer vision and speech recognition, the state-of-art methods are almost all deep neural networks. In the natural language processing (NLP) area, deep learning models are used to train a model that can represent words as vectors. Then researchers propose many deep learning models based on the word vectors for QA and summarization, etc. Convolutional neural networks (CNN) utilize filters to capture the local structures of the image, which performs very well on computer vision tasks. Researchers also find that CNN is effective on many NLP tasks. For instance, semantic parsing , sentence model, and other traditional NLP tasks.



III. PROBLEM DEFINITION

Given a set of m news articles containing the text and image information, we can represent the data as a set of text image tuples. In the fake news detection problem, we want to predict whether the news articles in A are fake news or not. We can represent the label set as Y = $\{[1,0],[0,1]\}$, where $[1,0]$ denotes real news while $[0,1]$ represents the fake news

IV. DATA ANALYSIS

To examine the finding from the raw data, a thorough investigation has been carried out to study the text and image information in news articles. There are some differences between real and fake news on American presidential election in 2016. We investigate the text and image information from various perspectives, such as the computational linguistic, sentiment analysis, psychological analysis and other image related features. We show the quantitative information of the data in this section, which are important clues for us to identify fake news from a large amount of data.

A. Dataset

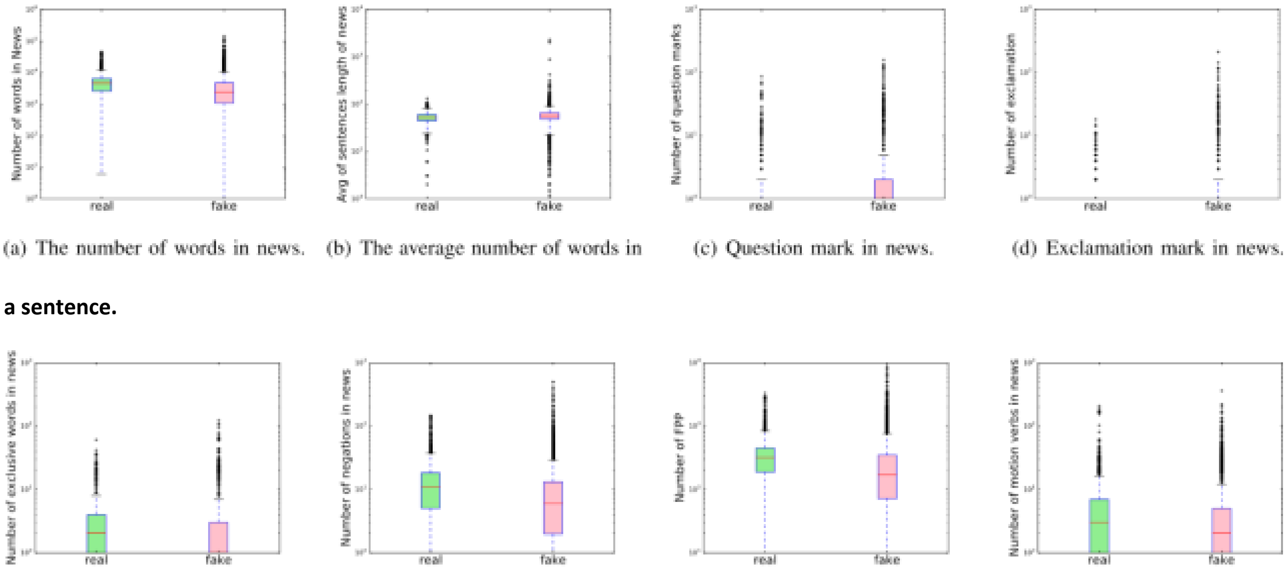
The dataset in this paper contains 20,015 news, i.e., 11,941 fake news and 8,074 real news. It is available on google drive[1]. If it is not available, you can also get the data set from one drive [2]. For fake news, it contains text and metadata scraped from more than 240 websites . The real news is crawled from the well known authoritative news websites, i.e., the New York Times, Washington Post, etc. The dataset contains multiple information, such as the title, text, image, author and website. To reveal the intrinsic differences between real and fake news, we solely use the title, text and image information.

B. Text Analysis

Let’s take the word frequency [23] in the titles as an example to demonstrate the differences between real and fake news in Fig. 2. If the news has no title, we set the title as ‘no title’. The frequently observed words in the title of fake news are *no title, IN, THE, CLINTON* and many meaningless numbers that represent special characters. We can have some interesting findings from the figure. Firstly, much fake news have no titles. These fake news are widely spread as the tweet with a few keywords and hyperlink of the news on social networks. Secondly, there are more capital characters in fake news. The purpose is to draw the readers’ attention, while the real news contains less capital letters, which is written in a standard format. Thirdly, the real news contain more detailed descriptions. For example, names (*Jeb Bush, Mitch McConnell*, etc.), and motion verbs (*left, claim, debate and poll*, etc.).

1) Computational Linguistic:

Number of words and sentences: Although liars have some control over the content of their stories, their underlying state of mind may leak out through the style of language used to tell the story. The same is true for the people who write the fake news. The data presented in the following paragraph provides some insight into the linguistic manifestations of this state of mind.



(a) The number of words in news.

(b) The average number of words in a sentence.

(c) Question mark in news.

(d) Exclamation mark in news.

(e) The exclusive words in news.

(f) The negations in news.

(g) FPP: First-person pronoun.

(h) Motion verbs in news.

fake news has fewer words than real news on average. There are 4,360 words on average for real news, while the number is 3,943 for fake news. Besides, the number of words in fake news distributes over a wide range, which indicates that some fake news have very few words and some have plenty of words. The number of words is just a simple view to analyse the fake news. Besides, real news has more sentences than fake news on average. Real news has 84 sentences, while fake news has 69 sentences. Based on the above analysis, we can get the average number of words in a sentence for real and fake news, respectively. As, the sentence of real news is shorter than that of fake news. Real news has 51.9 words on average in a sentence. However, the number is 57.1 for fake news. According to the box plot, the variance of the real news is much smaller than that of fake news. And this phenomenon appears in almost all the box plots. The reason is that the editor of real news must write the article under certain rules of the press. These rules include the length, word selection, no grammatical errors, etc. It indicates that most of the real news are written in a more standard and consistent way. However, most of the people who write fake news don't have to follow these rules.

***Question mark, exclamation and capital letters:* According to the statistics on the news text, real news has fewer question marks than fake news, The reasons may lie in that there are many rhetorical questions in fake news. These rhetorical questions are always used to emphasize the ideas consciously and intensify the sentiment.**

According to the analysis on the data, we find that both real and fake news have very few exclamations. However, the inner fence of fake news box plot is much larger than that of real news, as shown in Fig. 3(d). Exclamation can turn a simple indicative or declarative sentence into a strong command or reflect an emotional outburst. Hence, fake news is inclined to use the words with exclamations to fan specific emotions among the readers.

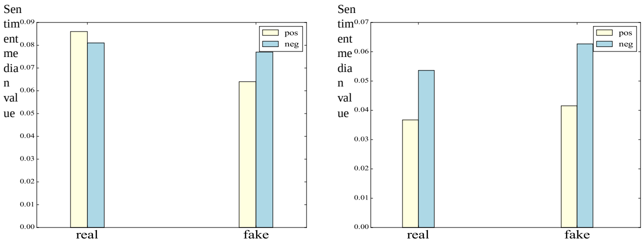
Capital letters are also analyse in the real and fake news. The reason for the capitalization in news is to draw readers attention or emphasize the idea expressed by the writers. According to the statistic data, fake news have much more capital letters than real news. It indicates that fake news deceivers are good at using the capital letters to attract the attention of readers, draw them to read it and believe it.

***Cognitive perspective:* From the cognitive perspective, we investigate the exclusive words (e.g., 'but', 'without', 'however') and negations (e.g.,, 'no', 'not') used in the news. Truth tellers use negations more frequently, The exclusive words in news have the similar phenomenon with the negations. The median of negations in fake news is much smaller than that of real news. The deceiver must be more specific and precise when they use exclusive words and negations, to lower the likelihood that being caught in a contradiction. Hence, they use fewer exclusive words and negations in writing. For the truth teller, they can exactly discuss what happened and what didn't happen in that real news writer witnessed the event and knew all the details of the event. Specifically, individuals who use a higher number of "exclusive" words are generally healthier than those who do not use these words .**

Psychology Perspective: **From the psychology perspective, we also investigate the use of first-person pronouns (e.g., I, we, my) in the real and fake news. Deceptive people often use language that minimizes references to themselves. A person who's lying tends not to use "we" and "I", and tend not to use person pronouns. Instead of saying "I didn't take your book," a liar might say "That's not the kind of thing that anyone with integrity would do" [31]. Similarly, as shown in Fig. 3(g), the result is the same with the point of view from the psychology perspective. On average, fake news has fewer first-person pronouns. The second-person pronouns (e.g., you, yours) and third-person pronouns (e.g., he, she, it) are also tallied up. We find that deceptive information can be characterized by the use of fewer first-person, fewer second-person and more third-person pronouns. Given space limitations, we just show the first-person pronouns figure. In addition, the deceivers avoid discussing the details of the news event. Hence, they use few motion verbs, as shown in Fig. 3(h).**

Lexical Diversity: **Lexical diversity is a measure of how many different words that are used in a text, while lexical density provides a measure of the proportion of lexical items (i.e. nouns, verbs, adjectives and some adverbs) in the text. The rich news has more diversity. According to the experimental results, the lexical diversity of real news is $2.2e\text{-}06$, which is larger than $1.76e\text{-}06$ for fake news.**

Sentiment Analysis: **The sentiment [26] in the real and fake news is totally different. For real news, they are more positive than negative ones. The reason is that deceivers may feel guilty or they are not confident to the topic. Under the tension and guilt, the deceivers may have more negative emotion [28], [35]. The experimental results agree with the above analysis in Fig. 4. The standard deviation of fake news on negative sentiment is also larger than that of real news, which indicates that some of the fake news have very strong negative sentiment.**



(a) The median sentiment values: (b) The standard deviation sentiment positive and negative. values: positive and negative.

Fig. 4. Sentiment analysis on real and fake news.

C. Image Analysis

We also analyse the properties of images in the political news. According to some observations on the images in the fake news, we find that there are more faces in the real news. Some fake news have irrelevant images, such as animals and scenes. The experiment result is consistent with the above analysis. There are 0.366 faces on average in real news, while the number is 0.299 in fake news. In addition, real news has a better resolution image than fake news. The real news has 457 × 277 pixels on average, while the fake news has a resolution of 355 × 228.

V. MODEL – THE ARCHITECTURE

In this section, we introduce the architecture of TI-CNN model in detail. Besides the explicit features, we innovatively utilize two parallel CNNs to extract latent features from both textual and visual information. And then explicit and latent features are projected into the same feature space to form new representations of texts and images. At last, we propose to fuse textual and visual representations together for fake news detection.

As shown in Fig. 5, the overall model contains two major branches, i.e., text branch and image branch. For each branch, taking textual or visual data as inputs, explicit and latent features are extracted for final predictions. To demonstrate the theory of constructing the TI-CNN, we introduce the model by answering the following questions: 1) How to extract the latent features from text? 2) How to combine the explicit and latent features? 3) How to deal with the text and image features together? 4) How to design the model with fewer parameters? 5) How to train and accelerate the training process?

A. Text Branch

For the text branch, we utilize two types of features: textual explicit features and textual latent features. The textual explicit features are derived from the statistics of the news text as we mentioned in the data analysis part, such as the length of the news, the number of sentences, question marks, exclamations and capital letters, etc. The statistics of a single news can be organized as a vector with fixed size. Then the vector is transformed by a fully connected layer to form a textual explicit features.

The textual latent features in the model are based on a variant of CNN. Although CNNs are mainly used in Computer Vision tasks, such as image classification [25] or object recognition [38], CNN also show notable performances in many Natural Language Processing (NLP) tasks [24], [46]. With the convolutional approach, the neural network can produce local features around each word of the adjacent word and then combines them using a max operation to create a fixed sized word-level embedding

$$\mathbf{X}^{Tl}_{1:n} = \mathbf{x}_{i,1} \oplus \mathbf{x}_{i,1} \oplus \mathbf{x}_{i,2} \oplus \dots \oplus \mathbf{x}_{i,n_i}, \quad .$$

It means that the news $\mathrm{X}^{Tl}_{i,1:n}$ is concatenated by each word. In this case, each news can be represented as a matrix. Then we use convolutional filters construct the new features. For instance, a window of words $\mathrm{X}^{Tl}_{i,j:j+h-1}$ can produce a feature c_i as follows:

$c_i = f(\mathrm{w} \cdot \mathrm{X}_{Tli,j:j+h-1} + b),$

where the $b \in R$ is the bias, and · is the convolutional operation. f is the non-linear transformation, such as the sigmoid and tangent function. A feature map is generated from the filter by going through all the possible window of words in the news.

$\mathrm{c} = [c_1, c_2, \dots, c_{n-h+1}],$

where $\mathrm{c} \in \mathrm{R}^{n-h+1}$. A max-pooling layer [30] is applied to take the maximum in the feature map c. The maximum value is denoted as $\hat{c} = max\{\mathrm{c}\}$. The max-pooling layer can greatly improve the robustness of the model by reserving the most important convolutional results for fake news detection. The pooling results are fed into a fully connected layer to obtain our final textual latent features for predicting news labels.

B. Image Branch

Similar to the text branch, we use two types of features: visual explicit features and visual latent features. In order to obtain the visual explicit features, we firstly extract the resolution of an image and the number of faces in the image to form a feature vector. And then, we transform the vector into our visual explicit feature with a fully connected layer.

Although visual explicit features can convey information of images contained in the news, it is hand-crafted features and not data-driven. To directly learn from the raw images contained in the news to derive more powerful features, we employ another CNN to learn from images in the news.

C. Rectified Linear Neuron

The sigmoid and tan activation functions may cause the gradient explode or vanishing problem [34] in convolutional neural networks. Hence, we add the activation to the image branch to the problem of gradient vanishing.

$$y = max(0, \sum_{i=1}^{k} x_i\theta_i + b)$$

Improve neural networks by speeding up training. The gradient computation is very simple (either 0 or 1 depending on the sign of x). Any negative elements are set to 0.0 – no exponentials, no multiplication or division operations.

Logistic and hyperbolic tangent networks suffer from the vanishing gradient problem, where the gradient essentially becomes 0 after a certain amount of training (because of the two horizontal asymptotes) and stops all learning in that section of the network.

D. Regularization

As shown in Table III, we dropout [40] as well as l_2-norms to prevent overfitting. Dropout is to set some of the elements in weight vectors as zero with a probability p of the hidden units during the forward and backward propagation. For instance, we have a dense layer and r is a vector where all the elements are zero. When we start to train the model, the dropout is to set some of the elements of r as 1 with probability as p. Suppose the output of dense layer is y. Then the dropout operation can be formulated as

$$y = \theta \cdot (z \circ r) + b,$$

where θ is the weight vector. $\circ$ is the element-wise multiplication operator. When we start to test the performance on the test dataset, the deleted neurons are back. The deleted weight are scaled by p such that $\hat{\theta} = p\,\theta$. The $\hat{\theta}$ is used to predict the test samples. The above procedure is implemented iteratively, which greatly improve the generalization ability of the model. We also use early stopping [36] to avoid overfitting. It can also be considered a type of regularization method (like L1/L2 weight decay and dropout) E. Network Training

We train our neural network by minimizing the negative likelihood on the training dataset D. To identify the label of a news X, the network with parameter θ computes a value $x)_{\tau}$. Then a sigmoid function is used over all the scores of tags $\tau \in T$ to transform the value into the conditional probability distribution of labels:

$$p(\tau|\mathbb{X}, \theta) = \frac{e^{s_\theta(\mathbb{X})_\tau}}{\sum_{\forall i \in T} e^{s_\theta(\mathbb{X})_i}}$$

We use the RMS prop [16] to minimize the loss function with respect to parameter θ:

(X,Y)∈D where X is the input data, and Y is the label of the news. We naturally choose back-propagation algorithm [15] to compute the gradients of the network structure. With the fine-tuned parameters, the loss converges to a good local minimum in a few epochs.

VI. EXPERIMENTS

A. Case study

A case study of the fake news is given in this section. The two fake news in Table II correspond to the Fig. 1(c) and 1(d). The first fake news is an article reporting that 'the American Amish Brotherhood endorsed Donald Trump for President'. However, the website is a fake CNN page. The image in the fake news can be easily searched online, and it is not very relevant with the news texts[3]. For the second fake news – 'Wiki leaks gave Hillary Clinton less than a 24-hour window to drop out of the race', it is actually not from Wiki leaks. Besides, the composite image [4] in the news is low quality.

B. Experimental Setup

We use 80% of the data for training, 10% of the data for validation and 10% of the data for testing. All the experiments are run at least 10 times separately. The textual explicit sub branch and visual explicit sub branch are connected with a dense layer. The parameters in these sub branches can be learned easily by the back-propagation algorithm. Thus, most of the parameters, which need to be tuned, exist in the textual latent sub branch and visual latent sub branch. The parameters are set as follows.

1) Text branch: For the textual latent sub branch, the embedding dimension of the word2vec is set to 100. The details of how to select the parameters are demonstrated in the sensitivity analysis section. The context of the word2vec is set to 10 words. The filter size in the convolutional neural network is $(3,3)$. There are 10 filters in all. Two dropouts are adopted to improve the model's generalization ability. For the textual explicit sub branch, we add a dense layer with 100 neurons first, and then add a batch normalization layer to normalize the activations of the previous layer at each batch, i.e. applies a transformation that maintains the mean activation close to 0 and the activation standard deviation close to 1. The outputs of textual explicit sub branch and textual latent feature sub branch are combined by summing the outputs up.

MODELS SPECIFICATIONS. BN: BATCH NORMALIZATION, RELU:

RECTIFIED LINEAR ACTIVATION FUNCTION, CONV: CONVOLUTIONAL

LAYER ON 2D DATA, CONV1D: CONVOLUTIONAL LAYER ON 1D DATA, DENSE: DENSE LAYER, EMB: EMBEDDING LAYER, MAXPO:

MAX-POOLING ON 2D DATA, MAXPO1D: MAX-POOLING ON 1D DATA.

2) Image branch: For the visual latent sub branch, all the images are reshaped as size (50×50). Three convolutional layers are added to the network hierarchically. The filters size is set to $(3,3)$, and there are 32 filters for each convolutional layer followed by a activation layer. A max pooling layer with pool size $(2,2)$ is connected to each convolutional layer to reduce the probability to be over-fitting. Finally, a flatten, batch normalization and activation layer is added to the model to extract the latent features from the images. For the explicit image feature sub branch, the input of the explicit features is connected to the dense layer with 100 neurons. And then a batch normalization and activation layer are added. The outputs of image convolutional neural network and explicit image feature sub branch are combined by summing the outputs up. We concatenate the outputs of text and image branch. An activation layer and dense layer are transforming the output into two dimensions. The labels of the news are given by the last sigmoid activation layer. In Table III, we show the parameter settings in the TI-CNN model. The total number of parameters is 7,509,980, and the number of trainable parameters is 7,509,176.

C. Experimental Results

We compare our model with several competitive baseline methods in Table IV. With image information only, the model cannot identify the fake news well. It indicates that image information is insufficient to identify the fake news. With text information, traditional machine learning method — logistic regression [18] is employed to detect the fake news. However, logistic regression fails to identify the fake news using the text information. The reason is that the hyperplane is linear, while the raw data is linearly inseparable. GRU [5] and Long short term memory [17] with text information are inefficient with very long sequences, and the model with 1000 input length performs worse. Hence, we take the input length 400 as the baseline method. With text and image information, TI-CNN outperforms all the baseline methods significantly.

THE EXPERIMENTAL RESULTS ON MANY BASELINE METHODS. THE

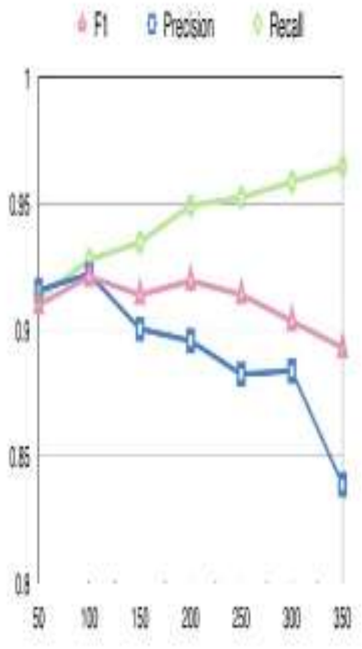
NUMBER AFTER THE NAME OF THE MODEL IS THE MAXIMUM INPUT

LENGTH FOR TEXTUAL INFORMATION. FOR THOSE NEWS TEXT LESS THAN $1{,}000$ WORDS, WE PADDED THE SEQUENCE WITH 0.

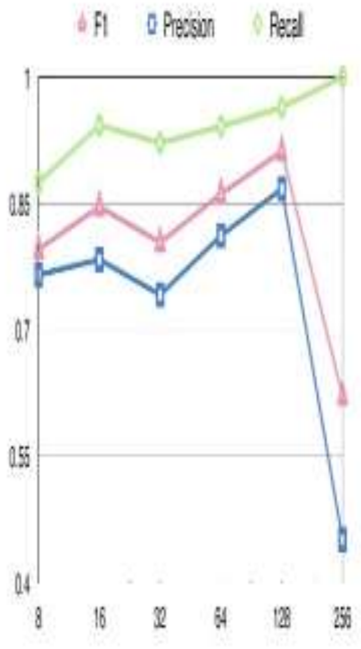
Method	Precision	Recall	F1-measure
CNN-image	0.5387	0.4215	0.4729
LR-text-1000	0.5703	0.4114	0.4780
CNN-text-1000	0.8722	0.9079	0.8897
LSTM-text-400	0.9146	0.8704	0.8920
GRU-text-400	0.8875	0.8643	0.8758
TI-CNN-1000	0.9220	0.9277	0.9210

D. Sensitivity Analysis

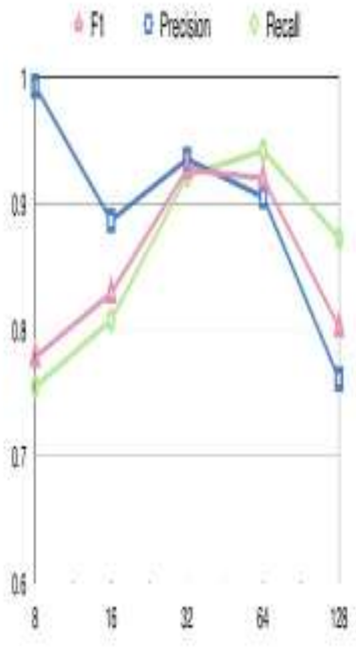
In this section, we study the effectiveness of several parameters in the proposed model: the word embedding dimensions, batch size, the hidden layer dimensions, the dropout probability and filter size.



(a) Word embedding dimension and F1-measure.



(b) Batch size and F1-measure.



(c) Hidden layer dimension and F1-measure.

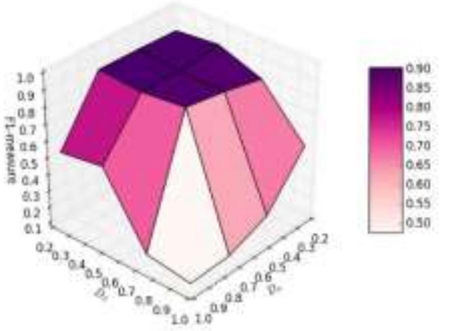
.

word embedding dimensions: In the text branch, we exploit a 3 layer neural network to learn the word embedding. The learned word vector can be defined as a vector with different dimensions, i.e., from 50 to 350. In Fig. 6(a), we plot the relation between the word embedding dimensions and the performance of the model. As shown in figure 6(a), we find that the precision, recall and f1-measure increase as the word embedding dimension goes up from 50 to 100. However, the precision and recall decrease from 100 to 350. The recall of the model is growing all the time with the increase of the word embedding dimension. We select 100 as the word embedding dimension in that the precision, recall and f1-measure are balanced. For fake news detection in real world applications, the model with high recall is also a good choice. The reason is that publishers can use high recall model to collect all the suspected fake news at the beginning, and then the fake news can be identified by manual inspection.

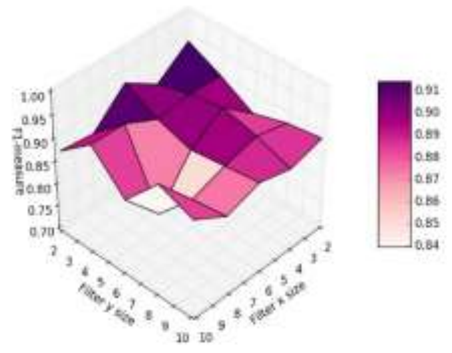
batch size: Batch size defines the number of samples that going to be propagated through the network. The higher the batch size, the more memory space the program will need. The lower the batch size, the less time the training process will take. The relation between batch size and the performance of the model is shown in Fig. 6(b). The best choice for batch size is 32 and 64. The F1 measure goes up from batch size 8 to 32 first, and then drops when the batch size increases from 32 to 128. For batch size 8, it takes 32 seconds to train the data on each epoch. For batch size 128, it costs more than 10

minutes to train the model on each epoch.

hidden layer dimension: As shown in Fig. 5, there are many hidden dense layers in the model. Deciding the number of neurons in the hidden layers is a very important part of deciding the overall neural network architecture. Though these layers do not directly interact with the external environment, they have a tremendous influence on the final output. Using too few neurons in the hidden layers will result in under fitting. Using too many neurons in the hidden layers can also result in several problems. Some compromise must be reached between too many and too few neurons in the hidden layers. As shown in Fig. 6(c), we find that 128 is the best choice for hidden layer dimension. The performance firstly goes up with the increase of the hidden layer dimension from 8 to 128. However, the



Dropout probabilities (D_α,D_β) and the performance of the model.



Filter size and the performance of the model.

Fig. 7. Dropout probabilities (D_α,D_β), filter size and the performance of the model.

dimension of the hidden layer reaches 256, the performance of the model drops due to overfitting.

d) Dropout probability and filter size: We analyse the dropout probabilities, as shown in Table III. D_α in Fig. 7(a) is the dropout layer connected to the text embedding layer, while D_β is used in both text and image branches. We use the grid search to choose the dropout probabilities. The model performs well when the D_α in the range [0.1,0.5] and the D_β in range [0.1,0.8]. In this paper, we set the dropout probabilities as (0.5,0.8), which can improve the model's generalization ability and accelerate the training process.

The filter size of a 1-dimension convolutional neural network layer in the textual latent sub branch is also a key factor in identifying the performance of the model. According to the paper [24], the model prefers small filter size for text information. It is consistent with the experimental results in Fig. 7(b). When the filter size is set to (3,3), the F1-measure of the model is 0.92-0.93.

CONCLUSIONS AND FUTURE WORK

The spread of fake news has raised concerns all over the world recently. These fake political news may have severe consequences. The identification of the fake news grows in importance. In this paper, we propose a unified model, i.e., TICNN, which can combine the text and image information with the corresponding explicit and latent features. The proposed model has strong expandability, which can easily absorb other features of news. Besides, the convolutional neural network makes the model to see the entire input at once, and it can be trained much faster than LSTM and many other RNN models. We do experiments on the dataset collected before the presidential election. The experimental results show that the TI-CNN can successfully identify the fake news based on the explicit features and the latent features learned from the convolutional neurons.

The dataset in this paper focuses on the news about American presidential election. We will crawl more data about the France national elections to further investigate the differences between real and fake news in other languages. It's also a promising direction to identify the fake news with much social network information, such as the social network structures and the users' behaviours. In addition, the relevance between headline and news texts is a very interesting research topic, which is useful to identify the fake news. As the development of Generative Adversarial Networks (GAN) [11], [37], the image can generate captions. It provides a novel way to evaluate the relevance between image and news text.

E. Network Training

We train our neural network by minimizing the negative likelihood on the training dataset D. To identify the label of a news X, the network with parameter θ computes a value sw(x)τ. Then a sigmoid function is used over all the scores of tags τ ∈ T to transform the value into the conditional probability distribution of labels:

The negative log likelihood of Equation 8

(X,Y)∈D where X is the input data, and Y is the label of the news. We naturally choose back-propagation algorithm [15] to compute the gradients of the network structure. With the fine-tuned parameters, the loss converges to a good local minimum in a few epochs.

VI. EXPERIMENTS

A. Case study

A case study of the fake news is given in this section. The two fake news in Table II correspond to the Fig. 1(c) and 1(d). The first fake news is an article reporting that 'the American Amish Brotherhood endorsed Donald Trump for President'. However, the website is a fake CNN page. The image in the fake news can be easily searched online, and it is not very relevant with the news texts . For the second fake news – 'Wiki leaks gave Hillary Clinton less than a 24-hour window to drop out of the race', it is actually not from Wiki leaks. Besides, the composite image in the news is low quality.

B. Experimental Setup

We use 80% of the data for training, 10% of the data for validation and 10% of the data for testing. All the experiments are run at least 10 times separately. The textual explicit sub branch and visual explicit sub branch are connected with a dense layer. The parameters in these sub branches can be learned easily by the back-propagation algorithm. Thus, most of the parameters, which need to be tuned, exist in the textual latent sub branch and visual latent sub branch. The parameters are set as follows.

Text branch:

For the textual latent sub branch, the embedding dimension of the word2vec is set to 100. The details of how to select the parameters are demonstrated in the sensitivity analysis section. The context of the word2vec is set to 10 words. The filter size in the convolutional neural network is (3,3). There are 10 filters in all. Two dropouts are adopted to improve the model's generalization ability. For the textual explicit sub branch, we add a dense layer with 100 neurons first, and then add a batch normalization layer to normalize the activations of the previous layer at each batch, i.e. applies a transformation that maintains the mean activation close to 0 and the activation standard deviation close to 1. The outputs of textual explicit sub branch and textual latent feature sub branch are combined by summing the outputs up.

TABLE II TWO FAKE NEWS CORRESPOND TO THE FIG. 1(C) AND 1(D).

Title	News text	Type
The Amish Brotherhood have endorsed Donald Trump for president.	The Amish, who are direct descendants of the protestant reformation sect known as the Anabaptists, have typically stayed out of politics in the past. As a general rule, they don't vote, serve in the military, or engage in any other displays of patriotism. This year, however, the AAB has said that it is imperative that they get involved in the democratic process.	Fake
Wiki leaks Gives Hillary An Ultimatum: QUIT, Or We Dump Something Life-Destroying	On Sunday, Wiki leaks gave Hillary Clinton less than a 24-hour window to drop out of the race or they will dump something that will destroy her "completely ."Recently, Julian Assange confirmed that WikiLeaks was not working with the Russian government, but in their pursuit of justice, they are obligated to release anything that they can to bring light to a corrupt system – and who could possibly be more corrupt than Crooked Hillary?	Fake

TABLE III

MODELS SPECIFICATIONS. BN: BATCH NORMALIZATION, RELU:

RECTIFIED LINEAR ACTIVATION FUNCTION, CONV: CONVOLUTIONAL

LAYER ON 2D DATA, CONV1D: CONVOLUTIONAL LAYER ON 1D DATA, DENSE: DENSE LAYER, EMB: EMBEDDING LAYER, MAXPO:

MAX-POOLING ON 2D DATA, MAXPO1D: MAX-POOLING ON 1D DATA.

THERE ARE TWO KINDS OF DROPOUT LAYERS, I.E., D = (Dα,Dβ), WHERE Dα = 0.5 AND Dβ = 0.8.

Text Branch Image Branch

Textual Explicit Textual Latent Visual

Latent Visual Explicit

Sigmoid

2) Image branch:

For the visual latent sub branch, all the images are reshaped as size (50 × 50). Three convolutional layers are added to the network hierarchically. The filters size is set to (3,3), and there are 32 filters for each convolutional layer followed by a activation layer. A max pooling layer with pool size (2,2) is connected to each convolutional layer to reduce the probability to be over-fitting. Finally, a flatten, batch normalization and activation layer is added to the model to extract the latent features from the images. For the explicit image feature sub branch, the input of the explicit features is connected to the dense layer with 100 neurons. And then a batch normalization and activation layer are added. The outputs of image convolutional neural network and explicit image feature sub branch are combined by summing the outputs up. We concatenate the outputs of text and image branch. An activation layer and dense layer are transforming the output into two dimensions. The labels of the news are given by the last sigmoid activation layer. In Table III, we show the parameter settings in the TI-CNN model. The total number of parameters is 7,509,980, and the number of trainable parameters is 7,509,176.

C. Experimental Results

We compare our model with several competitive baseline methods in Table IV. With image information only, the model cannot identify the fake news well. It indicates that image information is insufficient to identify the fake news. With text information, traditional machine learning method — logistic regression [18] is employed to detect the fake news. However, logistic regression fails to identify the fake news using the text information. The reason is that the hyperplane is linear, while the raw data is linearly inseparable. GRU [5] and Long short term memory [17] with text information are inefficient with very long sequences, and the model with 1000 input length performs worse. Hence, we take the input length 400 as the baseline method. With text and image information, TI-CNN outperforms all the baseline methods significantly.

TABLE IV

THE EXPERIMENTAL RESULTS ON MANY BASELINE METHODS. THE

NUMBER AFTER THE NAME OF THE MODEL IS THE MAXIMUM INPUT

LENGTH FOR TEXTUAL INFORMATION. FOR THOSE NEWS TEXT LESS THAN 1,000 WORDS, WE PADDED THE SEQUENCE WITH 0.

Method	Precision	Recall	F1-measure
CNN-image	0.5387	0.4215	0.4729
LR-text-1000	0.5703	0.4114	0.4780
CNN-text-1000	0.8722	0.9079	0.8897
LSTM-text-400	0.9146	0.8704	0.8920

GRU-text-400	0.8875	0.8643	0.8758
TI-CNN-1000	0.9220	0.9277	0.9210

D. Sensitivity Analysis

In this section, we study the effectiveness of several parameters in the proposed model: the word embedding dimensions, batch size, the hidden layer dimensions, the dropout probability and filter size.

(a) Word embedding dimension and F1-measure. (b) Batch size and F1-measure. (c) Hidden layer dimension and F1-measure.

Fig. 6. Word embedding dimension, batch size and the performance of the model.

a) word embedding dimensions: In the text branch, we exploit a 3 layer neural network to learn the word embedding. The learned word vector can be defined as a vector with different dimensions, i.e., from 50 to 350. In Fig. 6(a), we plot the relation between the word embedding dimensions and the performance of the model. As shown in figure 6(a), we find that the precision, recall and f1-measure increase as the word embedding dimension goes up from 50 to 100. However, the precision and recall decrease from 100 to 350. The recall of the model is growing all the time with the increase of the word embedding dimension. We select 100 as the word embedding dimension in that the precision, recall and f1-measure are balanced. For fake news detection in real world applications, the model with high recall is also a good choice. The reason is that publishers can use high recall model to collect all the suspected fake news at the beginning, and then the fake news can be identified by manual inspection.

b) batch size: Batch size defines the number of samples that going to be propagated through the network. The higher the batch size, the more memory space the program will need. The lower the batch size, the less time the training process will take. The relation between batch size and the performance of the model is shown in Fig. 6(b). The best choice for batch size is 32 and 64. The F1 measure goes up from batch size 8 to 32 first, and then drops when the batch size increases from 32 to 128. For batch size 8, it takes 32 seconds to train the data on each epoch. For batch size 128, it costs more than 10

minutes to train the model on each epoch.

c) hidden layer dimension: As shown in Fig. 5, there are many hidden dense layers in the model. Deciding the number of neurons in the hidden layers is a very important part of deciding the overall neural network architecture. Though these layers do not directly interact with the external environment, they have a tremendous influence on the final output. Using too few neurons in the hidden layers will result in under Fitting. Using too many neurons in the hidden layers can also result in several problems. Some compromise must be reached between too many and too few neurons in the hidden layers. As shown in Fig. 6(c), we find that 128 is the best choice for hidden layer dimension. The performance firstly goes up with the increase of the hidden layer dimension from 8 to 128. However, the

(a) Dropout probabilities (Dα,Dβ) and the performance of the model.

(b) Filter size and the performance of the model.

Fig. 7. Dropout probabilities (Dα,Dβ), filter size and the performance of the model.

dimension of the hidden layer reaches 256, the performance of the model drops due to overfitting.

d) Dropout probability and filter size: We analyse the dropout probabilities, as shown in Table III. Dα in Fig. 7(a) is the dropout layer connected to the text embedding layer, while Dβ is used in both text and image branches. We use the grid search to choose the dropout probabilities. The model performs well when the Dα in the range [0.1,0.5] and the Dβ in range [0.1,0.8]. In this paper, we set the dropout probabilities as (0.5,0.8), which can improve the model's generalization ability and accelerate the training process.

The filter size of a 1-dimension convolutional neural network layer in the textual latent sub branch is also a key factor in identifying the performance of the model. According to the paper [24], the model prefers small filter size for text information. It is consistent with the experimental results in Fig. 7(b). When the filter size is set to (3,3), the F1-measure of the model is 0.92-0.93.

VII. CONCLUSIONS AND FUTURE WORK

The spread of fake news has raised concerns all over the world recently. These fake political news may have severe consequences. The identification of the fake news grows in importance. In this paper, we propose a unified model, i.e., TICNN, which can combine the text and image information with the corresponding explicit and latent features. The proposed model has strong expandability, which can easily absorb other features of news. Besides, the convolutional neural network makes the model to see the entire input at once, and it can be trained much faster than LSTM and many other RNN models. We do experiments on the dataset collected before the presidential election. The experimental results show that the TI-CNN can successfully identify the fake news based on the explicit features and the latent features learned from the convolutional neurons.

The dataset in this paper focuses on the news about American presidential election. We will crawl more data about the France national elections to further investigate the differences between real and fake news in other languages. It's also a promising direction to identify the fake news with much social network information, such as the social network structures and the users' behaviours. In addition, the relevance between headline and news texts is a very interesting research topic, which is useful to identify the fake news. As the development of Generative Adversarial Networks (GAN) [11], [37], the image can generate captions. It provides a novel way to evaluate the relevance between image and news text.

Introduction

We consume news through several mediums throughout the day in our daily routine, but sometimes it becomes difficult to decide which one is fake and which one is authentic.

Do you trust all the news you consume from online media?

Every news that we consume is not real. If you listen to fake news it means you are collecting the wrong information from the world which can affect society because a person's views or thoughts can change after consuming fake news which the user perceives to be true.

Since all the news we encounter in our day-to-day life is not authentic, how do we categorize if the news is fake or real?

In this article, we will focus on text-based news and try to build a model that will help us to identify if a piece of given news is fake or real.

Before moving to the practical things let's get aware of few terminologies

Fake News

A sort of sensationalist reporting, counterfeit news embodies bits of information that might be lies and is, for the most part, spread through web-based media and other online media.

This is regularly done to further or force certain kinds of thoughts or for false promotion of products and is frequently accomplished with political plans.

Such news things may contain bogus and additionally misrepresented cases and may wind up being virtualized by calculations, and clients may wind up in a channel bubble.

In the document, words are present so many times that is called term frequency. In this section, if you get the largest values it means that word is present so many times with respect to other words. when you get word is parts of speech word that means the document is a very nice match.

IDF (Inverse Document Frequency): in a single document, words are present so many times, but also available so many times in another document also which is not relevant. IDF is a proportion of how critical a term is in the whole corpus.

Project

To get the accurately classified collection of news as real or fake we have to build a machine learning model.

To deals with the detection of fake or real news, After the first step is done, we will initialize the classifier, transform and fit the model. In the end, we will calculate the performance of the model using the appropriate performance matrix/matrices. Once will calculate the performance matrices we will be able to see how well our model performs.

The practical implementation of these tools is very simple and will be explained step by step in this article.

Let’s start.

Data Analysis

Here I will explain the dataset.

In this python project, we have used the CSV dataset. The dataset contains 7796 rows and 4 columns.

This dataset has four columns,

1. title: this represents the title of the news.

2. author: this represents the name of the author who has written the news.

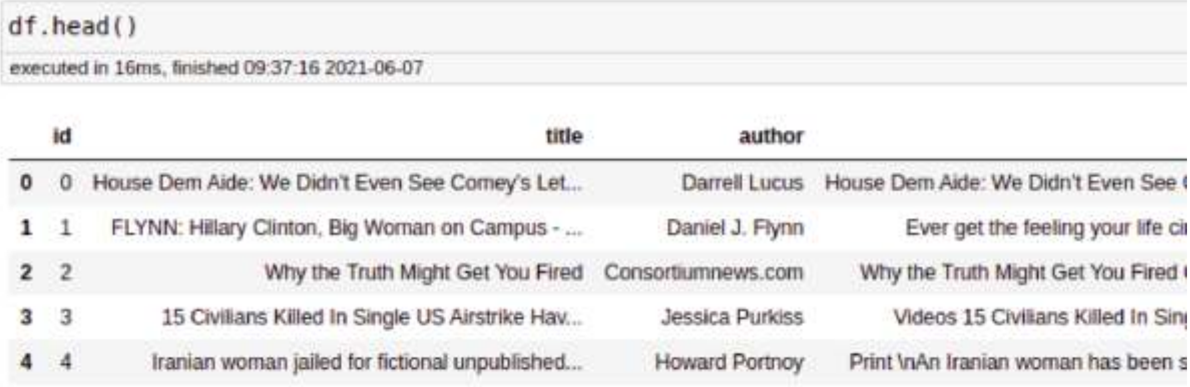
3. text: this column has the news itself.

4. label: this is a binary column representing if the news is fake (1) or real (0).

Libraries

The very basic data science libraries pandas and some specific libraries such as transformers.

Read dataset from CSV File



```
df.head()
executed in 16ms, finished 09:37:16 2021-06-07
```

	id	title	author	
0	0	House Dem Aide: We Didn't Even See Comey's Let...	Darrell Lucus	House Dem Aide: We Didn't Even See (
1	1	FLYNN: Hillary Clinton, Big Woman on Campus - ...	Daniel J. Flynn	Ever get the feeling your life ci
2	2	Why the Truth Might Get You Fired	Consortiumnews.com	Why the Truth Might Get You Fired (
3	3	15 Civilians Killed In Single US Airstrike Hav...	Jessica Purkiss	Videos 15 Civilians Killed In Sin
4	4	Iranian woman jailed for fictional unpublished...	Howard Portnoy	Print \nAn Iranian woman has been s

output:-

Before proceeding, we need to check whether a null value is present in our dataset or not.

There is no null value in this dataset. But if you have null values present in your dataset then you can fill it. In the code given below, I will tell you how you can replace the null values.

Data Pre processing

In data processing, we will focus on the text column on this data which actually contains the news part. We will modify this text column to extract more information to make the model more predictable. To extract information from the text column, we will use a library, which we know by the name of 'nl2tk'.

Here we will use functionalities of the 'nl2tk' library named Removing Stop words, Tokenization, and Lemmatization. So we will see these functionalities one by one with these three examples. Hope you will have a better understanding of extracting information from the text column after this.

Removing Stop words:-

These are the words that are used in any language used to connect words or used to declare the tense of sentences. This means that if we use these words in any sentence they do not add much meaning to the context of the sentence so even after removing the stop words we can understand the context.

Tokenization:-

Tokenization is the process of breaking text into smaller pieces which we know as tokens.

Each word, special character, or number in a sentence can be depicted as a token in NLP.

Tokenization is the process of breaking down a piece of code into smaller units called tokens.

CONVERTING LABELS:-

The dataset has a Label column whose datatype is Text Category. The Label column in the dataset is classified into two parts, which are denoted as Fake and Real. To train the model, we need to convert the label column to a numerical one.

VECTORIZATION

Vectorization is a methodology in NLP to map words or phrases from vocabulary to a corresponding vector of real numbers which is used to find word predictions, word similarities/semantics.

To make documents' corpora more relatable for computers, they must first be converted into some numerical structure. There are few techniques that are used to achieve this such as 'Bag of Words'.

More on that here.

MODELING

After Vectorization, we split the data into test and train data.

Logistic Regression

#LOGISTIC REGRESSION

Naive-Bayes

#NAIVE BAYES

Decision Tree

# DECISION TREE

Passive-Aggressive Classifier

Passive Aggressive is considered algorithms that perform online learning (with for example Twitter data). Their characteristic is that they remain passive when dealing with an outcome that has been correctly classified, and become aggressive when a miscalculation takes place, thus constantly self-updating and adjusting.

CONCLUSION

The passive-aggressive classifier performed the best here and gave an accuracy of

93.12%.

We can print a confusion matrix to gain insight into the number of false and true negatives and positives

Fake news detection techniques can be divided into those based on style and those based on content, or fact-checking. Too often it is assumed that bad style (bad

spelling, bad punctuation, limited vocabulary, using terms of abuse, ungrammaticality, etc.) is a safe indicator of fake news.

More than ever, this is a case where the machine's opinion must be backed up by clear and fully verifiable indications for the basis of its decision, in terms of the facts checked and the authority by which the truth of each fact was determined.

Collecting the data once isn't going to cut it given how quickly information spreads in today's connected world and the number of articles being churned out.

I hope you might find this helpful. You can comment down in the comment sections for any queries.