IEEE Access
Multidisciplinary : Rapid Review : Open Access Journal

Received 6 October 2023, accepted 16 October 2023, date of publication 19 October 2023, date of current version 27 October 2023.

Digital Object Identifier 10.1109/ACCESS.2023.3325895

APPLIED RESEARCH

Named Entity Recognition Utilized to Enhance Text Classification While Preserving Privacy

MOHAMMED KUTBI, (Member, IEEE)
Computer Science Department, Saudi Electronic University, Jeddah 23445, Saudi Arabia
e-mail: m.kutbi@seu.edu.sa

ABSTRACT Recent development in Natural Language Processing (NLP) techniques has encouraged NLP-based application in various field including business, legal and health. An important process for all NLP projects is text preprocessing which is a process that modifies text data before using them in a machine learning model. Usually text preprocessing process includes cleaning, filtering, removing and replacing some texts to increase model accuracy, robustness, reduce data size or preserve privacy. Named entities recognizer (NER) is an NLP tool which finds Named Entities in text such as: names, organization, addresses, numbers and date. In this work, we create a preproccessing approach that uses NER to find named entities and, then, replace them with their type i.e. location, person or organization name to improve accuracy and preserve privacy instead of removing them or letting them become noise to our data. Experiments for text classification task using our approach have been conducted on several datasets some of which were collected in-house. Experiments indicate that using this approach enhances classifier accuracy and reduces feature representation's dimensionality while, also, preserve privacy.

INDEX TERMS Named entities, preprocessing, text classification, privacy.

I. INTRODUCTION

Text data that is available now a days is presented in various forms such as: web pages, legal documents, micro blogs i.e. Twitter, online forums, online shopping stores such as amazon and eBay etc. Natural Language Processing (NLP) techniques have been improving for while and that inspired the creation of new NLP-based applications. NLP has been used in several tasks amongst which are: text classification [18], information extraction [11], information retrieval [29] and, natural language generation [12]. Most of these tasks require text preprocessing, which includes cleaning, filtering, removing and replacing some texts in the text data. Sometime, preprocessing handles only words with repeated characters or misspelled words, whereas in others only stemming and stop-words removal are used.

Named entities make up big part of text data. For instance, legal documents such as leases contain addresses, tenant name, landlord name, location and numbers as date, rent and fees. Moreover, for example a news article, include cities, addresses, people names, dates and organizations. Named Entity Recognition (NER) is used to find and classify named entities in a given text. There are extensive research on NER [9], [19]. Some of those developed NER model are targeting certain type of text such as biomedical data [2], [4], [10]. Not only the type of topic is relevant but also languages. Thus, there are some language specific NER models such as Portuguese [28], Arabic [33], English [3], [7] and Chinese [20]. Moreover, there are NER models specialized in a specific way of writing in a language such as historical documents [25]. Not necessarily NER model are designed so specific to a language, there are some NER model that are cross lingual by design [4], [16], [30]. Aside from the main purpose of NER models, which is finding and classifying named entities, they have been also used for privacy purpose such as [21] which, as part of the methodology, finds sensitive information in a text and tag them.

In this work, we attempt to make use of named entities instead of ignoring them. Our proposed approach works on two steps: (1) detecting, recognizing named entities, (2) replacing them with their category name in the original text. Using this approach, classifiers can benefit from named

The associate editor coordinating the review of this manuscript and approving it for publication was Nazar Zaki.

TABLE 1. Entity type and Description of SpaCy's ner system.

Entity type	description
PERSON	People, including fictional.
NORP	Nationalities or religious or political groups.
FAC	Facilities, such as buildings, airports, highways, bridges, etc.
ORG	Companies, agencies, institutions, etc.
GPE	Countries, cities, states.
LOC	Non-GPE locations, mountain ranges, bodies of water.
PRODUCT	Vehicles, weapons, foods, etc. (Not services).
EVENT	Named hurricanes, battles, wars, sports events, etc.
WORK OF ART	Titles of books, songs, etc.
LAW	Named documents made into laws.
LANGUAGE	Any named language.
DATE	Absolute or relative dates or periods.
TIME	Times smaller than a day.
PERCENT	Percentage.
MONEY	Monetary values, including unit.
QUANTITY	Measurements, as of weight or distance.
ORDINAL	"first", "second".
CARDINAL	Numerals that do not fall under another type.

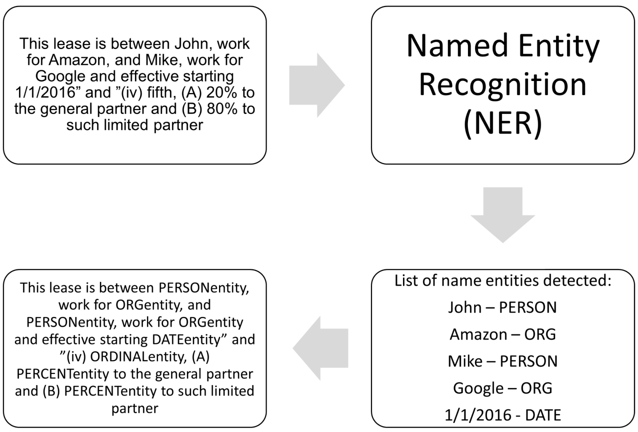

FIGURE 1. An overview of the proposed preprocessing approach.

entities in feature extraction and selection process plus prevent them from adding noise to our data. When using a feature selection methods such as Tfidf will remove named entities from features during the feature extraction process because named entities usually have low frequency unless many of training documents talk about the same topic/person. In case they do not have low frequency they will either help classifiers if the name relevant to the type of text e.g. "Obama" in politics or, become noise or have not effect, we will explain each cases in section III explicitly.

Potentially, this approach could be used to improve variety of NLP task from text classification to information retrieval in variety of fields such as legal documents or medical reports or even a social media micro blogs. We finds the legal domain of interest as privacy is very important while data are needed to be shared with machine learning experts for developing useful model. In our experiments, we applied our proposed approach on three different datasets. One of which is built in-house (legal documents). When comparing classifier performance with/without the use of our approach, we found that using our approach enhances classifiers' accuracy and slightly reduce feature dimensionality a.k.a data size. The main contribution of this paper are:

- A preprocessing approach which using Named Entities to enhance text classification performance in term of classification accuracy and dimensionality reduction.
- An in-house newly collected legal document dataset
- Showing that removing the all named entities to reserve privacy does not effect machine learning model performance negatively.

II. RELATED WORK

Named Entity Recognition (NER) is task of finding named entities in text and classifying them into predefined classes. NER system machine learning-based techniques that are trained to identify. NER has proven success in several fields such as medical [6]. There are some famous NER tools, which can be used such as Stanford-NER [7], SpaCy's NER [1] and NLTK NER [3]. Stanford-NER [7] is a JAVA tool developed by Stanford University team. It's based on Conditional Random Fields (CRF) and provides pre-trained models for extracting person, organization, location, and other entities. SpaCy's NER [1] is a Python framework that is an extremely fast statistical entity recognition system which assigns labels to contiguous spans of tokens. Also, it has several pre-trained model that can easily customized. NLTK NER [3] has a wrapper for the Stanford-NER [7] plus their own NER. NLTK NER [3] works over three steps; (1) Tokenising Words, (2) Part-Of-Speech (POS) (3) Chunking on POS. We used SpaCy's NER [1] in this paper.

There are some popular preprocessing techniques such as stop-word removal and stemming. Stop-words removal technique [31] detect and remove words that do not help in the targeted task and increase feature vector dimensionality. For instance, the words "is, "are" and "the" are stop words, in text classification, they do not indicate the type of paragraph, document or its sentiment. Thus, the Stop-word removal technique works on removing all words in the pre-defined stop-word list from text data. A very popular stop word list is SMART stop word list [26]. Another preprocessing technique is Stemming [14] which is used to find out the root of a word and converts every word in a text to their roots. The idea behind stemming is that words with the same root usually describe same or similar concepts in text. For example, the words, "worker", "workers", "worked", "work" and "working" can be stemmed to the word "work". The Porter Steamer algorithm [24] is a popular algorithm. Stemming has its advantages and disadvantages, it depends on the domain and the document but it does not always improve performance [13]. In this work, we did not use Stemming as we want to make the comparison between the use of our proposed method and not using very simple and fair.

Text classification is a well-studied task over long period and in many fields. Text classification has two parts: feature representation and classification. BERT [17] is a popular method for representing text in a feature space which could be used for different NLP tasks. BERT does not require process labelled text (raw text) to learn how to create representation which a big advantage. It has been evaluated on several NLP tasks and has shown impressive capabilities. In this paper, we have not used BERT due to its computational requirement.

Bag-of-word [32] is another famous method that has been around for awhile. It is designed to represent text in a feature space for later analysis. Bag-of-word simply creates a feature dimension for each word in find in the training data. Then, text data repeated words determine the value at each of those dimensions. A modified version of Bag-of-word is Tfidf [15] which is designed to improve feature selection by calculating their frequency. For simplicity, we used bag-of-word in our evaluation and Support Vector Machine (SVM) [5] for the classifier.

More et. al [22] have used some type of Named Entity Recognition to enhance preprocessing in the developed system. The system they developed was to find precedent legal cases from a dataset. This is similar to ours in the use of named entities and the domain of their dataset which is, also, legal. However, they have not studied the effect of the removal of named entities on text classification or on the finder system they developed.

III. PURPOSED APPROACH

Our purposed approach is divided into two parts, finding and classifying name entities and, then, replacing them with class name in the original text. In this section, we will present the proposed approach.

A. DETECT AND CLASSIFY NAME ENTITIES

For this task, we use, off-the-shelf, SpaCy NER software which is available on spacy.io website [1]. We use SpaCy NER API to pass text to the NER. Then, it returns a list of named entities and their type which are found in the inputed text. SpaCy NER uses tagging and dependency parsing to detect the names and classify them using a name trained classifier. Table 1 has a list of all types of named entities that SpaCy NER detects and classify: Here is an example of sentence and NER result, which also shown in Fig1:

"This lease is between John, work for Amazon, and Mike, work for Google and effective starting 1/1/2016"

List of name entities detected and classified is follow:

- John - PERSON
- Amazon - ORG
- Mike - PERSON
- Google - ORG
- 1/1/2016 - DATE

Another example:

"(iv) fifth, (A) 20% to the general partner and (B) 80% to such limited partner." List of name entities detected and classified is follow:

- fifth - ORDINAL
- 20% - PERCENT
- 80% - PERCENT

B. REPLACE NAME ENTITY WITH LABEL

Here we describe an algorithm to search for each detected entity from the NER system and replace it with named entity type provided by the NER. Also, we concatenate the type of named entity with string "entity" to avoid confusion such as the type "PERSON" for instance. Algorithm1 contains detailed script. Here is an example of a string passed to the algorithm as an input and the algorithm outputs, which is also shown in Fig 1:

- Input: "This lease is between John, work for Amazon, and Mike, work for Google and effective starting 1/1/2016" and "(iv) fifth, (A) 20% to the general partner and (B) 80% to such limited partner."
- Output: "This lease is between PERSONentity, work for ORGentity, and PERSONentity, work for ORGentity and effective starting DATEentity" and "(iv) ORDINALentity, (A) PERCENTentity to the general partner and (B) PERCENTentity to such limited partner".

C. TEXT CLASSIFICATION

To better understand the impact of replacing named entites on text classification task as an example. Text classification is the task of classifying text into one of predefined classes. To be more specific, the task is classifying paragraphs based on what information they contain. So it makes information extraction on specific paragraph or paragraphs instead of whole document.

We will evaluate the performance of a text classifier when our preprocessing step is used or not. Performance of both with and without our proposed preprocessing approach will be reported in term of accuracy of the classification task. To make this compression fair we use the same setting, algorithm, vectorization method in all experiments. We used

TABLE 2. This table contains all text classes and number of instance in each class in the dataset.

Class	No. of instances
calendar business days	4
carried interest percent	9
change of control	30
fmv determination mechanic	6
fund counsel	11
general partner name	13
clawback at end of fund	14
giveback	12
mgmt fee during invest period pct OF ART	11
mgmt fee return before the carry	21
mgmt fees inside commitment	9
number of days	25
preferred return percent	15
restricted	34
right of refusal	19
right to terminate investment period with cause	6
stepdown in fees upon successor	19
tax domicile	10
transfer right of first refusal	21
waterfall type	30
Total	**319**

bag-of-word as feature extractor method and SVM as our classifier. To create the feature space there are options of number grams and the range. Thus we experiment with different number of n-gram ranges (1,1), (1,2), (1,3) to find if there is a difference in performance. We used 5 fold-cross-validation method to experiment with different data distributions and repeated experiment 10 times. Thus, all reported number will be the average of 10 repeated accuracy results.

Algorithm 1 Proposed Approach Algorithm

procedure ProposedApproach(t)
  Entities=SpaCy'sNER(t) ▷ Entities=list(text,label)
  for each E in Entities **do**
    $t = t$.replace(E.text, E.label+"entity")
  end for
  return t
end procedure

IV. EXPERIMENT

In our experiment, we focus on text classification, an NLP task. Our goal is to evaluate our proposed approach in a text classification task and compare result with/without using our preprocessing approach. It is worth noting that we do not compare our method performance with other preprocessing methods because it is not meant to replace them but to complement other preprocessing steps.

A. DATASETS

In this experiment, we use three datasets, one of legal document and two of non-legal document to verify our approach on other domains specifically on a in-house collected dataset and two public datasets. Our first experiment is on a dataset consist of legal documents called Limited Partnership Agreement. Then, two more experiments on movie reviews, and comparative sentences.

1) LIMITED PARTNERSHIP AGREEMENT (LPA)

This dataset was collected in-house. LPA dataset consists of 20 classes of total of 319 paragraphs, which specified in details in Table 2. Class name represents the information a paragraph contains. For instance if paragraph contains carried interest percent then its label is carried interest percent. The main goal for classification here is to facilitate information retrieval as next step, out of scope of this paper.

2) MOVIE REVIEWS (MR)

MR dataset [23], [27] consist of 2 classes, postive and negtive, of total of 2000 reviews, which specified in details in Table 3. Class name represents the sentiment of the review. For instance if a review contains positive sentiment then its label is positive and vice versa. This dataset designed to train a classifier that can classify review into position/negative which very useful for the business perspective.

3) COMPARATIVE SENTENCES (CS)

CS dataset [8] consist of 4 classes, three comparative classes and non-comparative, of total of 853 sentences, which specified in details in Table 4. Class name represents the type of sentence. For instance if a sentence is not a comparative sentence then its label is non-comparative. The four classes are described Ganapathibhotla et. al. [8] as follow:"

- 1) Non-equal gradable: Relations of the type greater or less than that express a total ordering of some entities with regard to their shared features. For example, the sentence, "Camera X's battery life is longer than that of Camera Y", orders "Camera X" and "Camera Y" based on their shared feature "battery life".

TABLE 3. This table contains all classes and number of instance in Movie reviews dataset.

Class	No. of instances
positive	1000
negative	2000
Total	**3000**

TABLE 4. This table contains all classes and number of instance in Movie reviews dataset.

Class	No. of instances
Non-equal gradable	362
Equative	170
Superlative	174
Non-gradable	147
Total	**853**

- 2) Equative: Relations of the type equal to that state two objects as equal with respect to some features, e.g., "Camera X and Camera Y are about the same size".
- 3) Superlative: Relations of the type greater or less than all others that rank one object over all others, "Camera X's battery life is the longest".
- 4) Non-gradable: Sentences which compare features of two or more entities, but do not explicitly grade them, e.g., "Camera X and Camera Y have different features""

B. RESULT ANALYSIS

To test the success of the proposed method, we are training and evaluating text classifiers on each the above datasets with/out applying the proposed method. To measure the success of the proposed method, we are looking at four metrics: (1) Performance of the text classifier in term of accuracy which is the ratio of classifying text correctly, (2) Dimensionality of the feature vector representing a data sample. (3) Privacy of the Named Entities.

V. RESULTS AND DISCUSSION

In this section, we present the experiments' results using proposed approach and without. Also, we discuss some effect on performance based on the type of dataset.

A. PERFORMANCE

Here we measure performance of text classifier in term of accuracy. Reported accuracy in this experiment is averaged of 10 experiments on each setting. Experiments show that accuracy increased significantly in legal document experiment and slight increased in comparative sentences and movie reviews analysis. Table 5 contains results of accuracy for both with and without our approach combined with different n-gram ranges. See V-D for the discussion of this point.

B. DIMENSIONALITY

Our proposed approach reduces the dimensionality of the feature space which expected as it reduce all names of people to one string "personEntity" for instance. Table 6 contains the size of feature vectors for each setting and with and without using our setting.

TABLE 5. accuracy of SVM classifier with and without proposed approach. NE stands for Named Entity.

Dataset	Method	1,1	1,2	1,3
LPA	Linear SVM	55.37	55.60	57.48
LPA	Linear SVM + NE	58.31	63.76	62.83
MR	Linear SVM	81.00	82.50	82.60
MR	Linear SVM + NE	81.90	82.80	82.50
CS	Linear SVM	68.30	66.66	65.25
CS	Linear SVM + NE	68.54	67.37	66.59

TABLE 6. Vector size with and without purposed approach. NE stands for Named Entity.

Dataset	Method	1,1	1,2	1,3
LPA	Linear SVM	1966	12299	30646
LPA	Linear SVM + NE	1927	12210	30683
MR	Linear SVM	29713	317524	837155
MR	Linear SVM + NE	29515	313633	831455
CS	Linear SVM	2301	9096	17093
CS	Linear SVM + NE	2091	8505	16330

Feature extraction approach that is used in almost all NLP methods is bag-of-word. Some methods use Tfidf to deal high frequency words, stop words, low frequency words, name entities and. Using Tfidf will result in deleting most or all named entities since they will be of low frequency and bag-of-word without Tfidf will result in keeping them and this will result in higher dimension feature vector and noise. Our Approach will eliminate the noise caused by having different values such as, documentX has carried interest rate of 10% and documentY has carried interest rate of 20%. In some text classification problem, like 20 newsgroups [12], named entities are useful to the classifier because names in a group most likely to appear in that same group only such as if we are classifying news into categories, named entity "Obama" will appear only in politic news. Our approach is likely to benefits applications/datasets where actual named entities are irrelevant but knowing they are existing in a text is important. Our approach save memory space and enhance performance accuracy.

C. PRIVACY

An important component in any machine learning-based system is data. Sometimes such as in legal context those are very secretive. Thus, it is hard to train machine-learning model for legal content without exposing the legal content to the developing team which sometime even works for a third party. Using this method, we can mask all named entity before sharing the data with the developing team without sacrificing about exposing privacy.

D. RATIONAL ON THE PERFORMANCE RESULTS

Our approach will increase the probability of classifying a paragraph that contains certain value type when we are looking for a paragraph that must contain it regardless of its value. For instance, we are classifying a paragraph that contains percentage value, our approach is more likely to classify it as one of the classes that must contain a percentage value such as carried interest rate. Accuracy is, as expected, increased when the actual named entity does not help in classification but their existence does such as in the LPA dataset where the type of paragraph is not determined by names but by whether a name type exist or not. On the other hand, movie reviews maybe related to an actor name or organization name when they are in a movie it influence movie reviews' writers opinions. Thus, we see the accuracy did not increase and sometime decreased when we used or approach.

REFERENCES

[1] M. Honnibal and I. Montani, ''spaCy 2: Natural language understanding with Bloom embeddings, convolutional neural networks and incremental parsing,'' 2017.

[2] P. Bhatia, B. Celikkaya, M. Khalilia, and S. Senthivel, ''Comprehend medical: A named entity recognition and relationship extraction web service,'' in *Proc. 18th IEEE Int. Conf. Mach. Learn. Appl. (ICMLA)*, Dec. 2019, pp. 1844–1851.

[3] S. Bird, E. Klein, and E. Loper, *Natural Language Processing With Python: Analyzing Text With the Natural Language Toolkit*. Sebastopol, CA, USA: O'Reilly Media, 2009.

[4] R. Catelli, F. Gargiulo, V. Casola, G. De Pietro, H. Fujita, and M. Esposito, ''Crosslingual named entity recognition for clinical de-identification applied to a COVID-19 Italian data set,'' *Appl. Soft Comput.*, vol. 97, Dec. 2020, Art. no. 106779.

[5] C. Cortes and V. Vapnik, ''Support-vector networks,'' *Mach. Learn.*, vol. 20, no. 3, pp. 273–297, Jul. 1995.

[6] S. Dingare, M. Nissim, J. Finkel, C. Manning, and C. Grover, ''A system for identifying named entities in biomedical text: How results from two evaluations reflect on both the system and the evaluations,'' *Comparative Funct. Genomics*, vol. 6, nos. 1–2, pp. 77–85, 2005.

[7] J. R. Finkel, T. Grenager, and C. Manning, ''Incorporating non-local information into information extraction systems by Gibbs sampling,'' in *Proc. 43rd Annu. Meeting Assoc. Comput. Linguistics (ACL)*, 2005, pp. 363–370.

[8] M. Ganapathibhotla and B. Liu, ''Mining opinions in comparative sentences,'' in *Proc. 22nd Int. Conf. Comput. Linguistics (COLING)*, 2008, pp. 241–248.

[9] A. Goyal, V. Gupta, and M. Kumar, ''Recent named entity recognition and classification techniques: A systematic review,'' *Comput. Sci. Rev.*, vol. 29, pp. 21–43, Aug. 2018.

[10] M. Gridach, ''Character-level neural network for biomedical named entity recognition,'' *J. Biomed. Informat.*, vol. 70, pp. 85–91, Jun. 2017.

[11] R. Grishman, ''Information extraction,'' *IEEE Intell. Syst.*, vol. 30, no. 5, pp. 8–15, Sep. 2015.

[12] M. A. Haidar and M. Rezagholizadeh, ''TextKD-GAN: Text generation using knowledge distillation and generative adversarial networks,'' in *Proc. 32nd Can. Conf. Artif. Intell., Can. AI*, Kingston, ON, Canada. New York, NY, USA: Springer, May 2019, pp. 107–118.

[13] A. F. Hidayatullah, C. I. Ratnasari, and S. Wisnugroho, ''Analysis of stemming influence on indonesian tweet classification,'' *TELKOMNIKA, Telecommun. Comput. Electron. Control*, vol. 14, no. 2, pp. 665–673, 2016.

[14] A. G. Jivani, ''A comparative study of stemming algorithms,'' *Int. J. Comp. Tech. Appl*, vol. 2, no. 6, pp. 1930–1938, 2011.

[15] T. Joachims, ''A probabilistic analysis of the Rocchio algorithm with TFIDF for text categorization,'' in *Proc. ICML*, vol. 97, 1997, pp. 143–151.

[16] K. Karthikeyan, Z. Wang, S. Mayhew, and D. Roth, ''Cross-lingual ability of multilingual bert: An empirical study,'' in *Proc. Int. Conf. Learn. Represent.*, 2020. [Online]. Available: https://cogcomp.seas.upenn.edu/papers/KWMR20.pdf

[17] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, ''BERT: Pre-training of deep bidirectional transformers for language understanding,'' in *Proc. NAACL-HLT*, 2019, pp. 4171–4186.

[18] K. Kowsari, K. J. Meimandi, M. Heidarysafa, S. Mendu, L. Barnes, and D. Brown, ''Text classification algorithms: A survey,'' *Information*, vol. 10, no. 4, p. 150, Apr. 2019.

[19] J. Li, A. Sun, J. Han, and C. Li, ''A survey on deep learning for named entity recognition,'' *IEEE Trans. Knowl. Data Eng.*, vol. 34, no. 1, pp. 50–70, Jan. 2022.

[20] P. Liu, Y. Guo, F. Wang, and G. Li, ''Chinese named entity recognition: The state of the art,'' *Neurocomputing*, vol. 473, pp. 37–53, Feb. 2022.

[21] F. Martinelli, F. Marulli, F. Mercaldo, S. Marrone, and A. Santone, ''Enhanced privacy and data protection using natural language processing and artificial intelligence,'' in *Proc. Int. Joint Conf. Neural Netw. (IJCNN)*, Jul. 2020, pp. 1–8.

[22] R. More, J. Patil, A. Palaskar, and A. Pawde, ''Removing named entities to find precedent legal cases,'' in *Proc. 11th Forum Inf. Retr. Eval. (FIRE)*, Kolkata India. New York, NY, USA: Association for Computing Machinery, Dec. 2019.

[23] B. Pang and L. Lee, ''Seeing stars: Exploiting class relationships for sentiment categorization with respect to rating scales,'' 2005, *arXiv:cs/0506075*.

[24] M. F. Porter, ''An algorithm for suffix stripping,'' *Program*, vol. 14, no. 3, pp. 130–137, Mar. 1980.

[25] A. C. Rouhou, M. Dhiaf, Y. Kessentini, and S. B. Salem, ''Transformer-based approach for joint handwriting and named entity recognition in historical document,'' *Pattern Recognit. Lett.*, vol. 155, pp. 128–134, Mar. 2022.

[26] G. Salton, *Automatic Text Processing: The Transformation, Analysis, and Retrieval of*. Reading, MA, USA: Addison-Wesley, 1989, p. 169.

[27] R. Socher, A. Perelygin, J. Wu, J. Chuang, C. D. Manning, A. Y. Ng, and C. Potts, ''Recursive deep models for semantic compositionality over a sentiment treebank,'' in *Proc. Conf. Empirical Methods Natural Lang. Process.*, 2013, pp. 1631–1642.

[28] F. Souza, R. Nogueira, and R. Lotufo, ''Portuguese named entity recognition using BERT-CRF,'' 2019, *arXiv:1909.10649*.

[29] N. Thakur, N. Reimers, A. Rücklé, A. Srivastava, and I. Gurevych, ''BEIR: A heterogenous benchmark for zero-shot evaluation of information retrieval models,'' 2021, *arXiv:2104.08663*.

[30] Q. Wu, Z. Lin, G. Wang, H. Chen, B. F. Karlsson, B. Huang, and C.-Y. Lin, ''Enhanced meta-learning for cross-lingual named entity recognition with minimal resources,'' in *Proc. AAAI Conf. Artif. Intell.*, vol. 34, 2020, pp. 9274–9281.

[31] X.-B. Xue and Z.-H. Zhou, ''Distributional features for text categorization,'' *IEEE Trans. Knowl. Data Eng.*, vol. 21, no. 3, pp. 428–442, Mar. 2009.

[32] Y. Zhang, R. Jin, and Z.-H. Zhou, ''Understanding bag-of-words model: A statistical framework,'' *Int. J. Mach. Learn. Cybern.*, vol. 1, nos. 1–4, pp. 43–52, Dec. 2010.

[33] A. Zirikly and M. Diab, ''Named entity recognition for Arabic social media,'' in *Proc. 1st Workshop Vector Space Model. Natural Lang. Process.*, 2015, pp. 176–185.

MOHAMMED KUTBI (Member, IEEE) received the B.S. degree from the Department of Information Technology, King Abdulaziz University, and the M.S. and Ph.D. degrees from the Department of Computer Science, Stevens Institute of Technology. He is currently an Assistant Professor with the Department of Computer Science, Saudi Electronic University (SEU). His recent work includes motion segmentation, domain adaptation, and human–robot interaction. His research interests include computer vision and robotics.