34th International Conference on Tool with Artificial Intelligence, ICTAI 2022 ,
31 Oct.-2 Nov. 2022, ICTAI 2022 , 31 Oct.-2 Nov. 2022, Virtual Event

Context–aware Relation Classification based on Deep Learning

Maha Mallek
LIS UMR CNRS 7020-LARIA, ENSI
University of Manouba - AMU
Manouba, Tunisia - Marseille, France
maha.mallek@lis-lab.fr

Ramzi Guetari
SERCOM laboratory
Polytechnic School of Tunisia
University of Carthage
La Marsa, Tunisia
ramzi.guetari@ept.ucar.tn

Sébastien Fournier
LIS UMR CNRS 7020
Aix Marseille University (AMU)
Marseille, France
sebastien.fournier@lis-lab.fr

Wided Lejouad Chaari
LARIA, ENSI
University of Manouba
Manouba, Tunisia
wided.chaari@ensi-uma.tn

Bernard Espinasse
LIS UMR CNRS 7020
Aix Marseille University (AMU)
Marseille, France
bernard.espinasse@lis-lab.fr

***Abstract*—Modern information supports carry heterogeneous data, in such large quantities that the traditional means of processing become obsolete and inefficient to meet today's needs. In addition to the quantity of data, the unstructured nature of this data requires new intelligent, efficient and automated processing techniques. In order to produce automatic systems capable of managing this data and extracting relevant knowledge from it, a number of problems must be solved, including the extraction and classification of relations from textual data. While the extraction of relations is mainly based on syntactic aspects of the text, the classification requires a semantic approach. Such existing relation classification systems deal only with few pre–defined types. These systems don't take into account the context, thus reducing the relevance of this classification. In this paper, we propose a simplified definition of what is context and, based on this definition, we propose an approach to classify relations according to their types while taking into account this context. The system, allowing to obtain a degree of "contextualization" of relations, has been tested on the SemEval–2010 Task–8, New York Times corpora and a contextual dataset, named WikiContext, that we have built for this purpose. The results show that our system outperforms the state–of–the–art relation classification systems, thus demonstrating the relevance of taking context into account in this classification process.**

***Index Terms*—Relations Classification, Context Identification, Markov Model, LSTM Network, Language Modeling.**

I. Introduction

We live in an era where almost everything is ruled by data. We have gone through the database, knowledge base and information highway trends to arrive TO the Big Data era. The volume of data is steadily and dramatically increasing year after year. This amount of data usually conveys vital information whose exploitation requires deep analysis and represents a tedious or even impossible task for human beings and the classic IT methods of data processing. More intelligent and efficient solutions are therefore needed.

Automatic data processing is a vast undertaking that consists of many things: acquisition, storage, search, retrieval, classification, etc. The origins of this data are diverse and varied and their nature and formats are not always the same. A multitude of specialized applications are then necessary to address effectively the various problems faced. Among the major concerns of our days, some stand out in particular: Information Extraction (IE) and decision support. The usefulness of IE does not limited to the decision support, but is also used in many other tasks such as information retrieval, question answering, etc.

Most of the data that passes through the Internet is textual, because this format is much less constraining than others, especially in terms of storage space and bandwidth consumption. The complexity of automatic processing of this data can vary considerably. When the data is structured, this task is quite simple, however, the effort required to manage unstructured textual data is considerable. Identifying and classifying relations in textual documents is one of the ongoing research areas. The extraction of relations allows the automation of information processing, and allows the discovery of semantic relations between entities in the raw text. Several approaches have been experimented with more or less convincing results. To address the problem, many researchers have proposed various systems and approaches, generally using supervised methods, which can be based on either human–designed rules [1], on logical relational learning as inductive logic programming [2], on statistical machine learning using feature–based or kernel based methods [3] or, more recently, based on deep learning.

A lot of works in relation classification, concern the use of deep leaning including Convolutional Neural Networks (CNN) [4], Recurrent Neural Networks (RNN) [5], and Long Short–Term Memory (LSTM) networks [6]. Several other authors, such as [7], have proposed other Deep Neural Network (DNN) models to classify relations. Regardless of the methods used, these approaches classify relations into a number of predefined types, essentially based on syntactic aspects of the text. These methods neglect the semantic aspects and can be misleading during classification. A pillar of semantics is to take into account the context in the classification of relations. This is precisely the subject we address in this paper and to which we propose some solutions.

The rest of the paper is organized as follows: in section II, we present related work on relation classification from text documents. Section III presents our proposed approach to relation classification. Section IV presents the experimental results of our approach on two reference corpora, SemEval 2010 and New York Time, as well as a specially designed "Wikicontext" dataset. In this section, we compare the performance of our approach to the results obtained by state–of–the–art relation classification systems. Finally, Section V concludes this paper and presents future work.

II. Related Work

There are three main approaches to relations extraction and classification using machine learning: supervised, semi–supervised, and unsupervised approaches. The supervised approach of relations classification is the predominant one. It is based on symbolic (logical) methods, statistical methods or deep learning methods.

Symbolic methods are mainly based on inductive logic programming [2]. In statistical based–methods, approaches are divided into Feature–based and Kernel–based methods. Feature–based methods identify a set of features from the textual content in order to classify the relations. In [8], Kambhatla identifies features of lexical, syntactic, and semantic nature. Indeed, he uses maximum entropy models to combine various text–derived features to predict the type of relation. Zhou et al. [9], based on the work of Kambhatla, investigated the integration of various lexical, syntactic, and semantic information into feature–based models using the support vector machine (SVM) model. They demonstrated that some of the important features, such as WordNet, can be used in feature–based models to further improve the performance of relation classification.

Kernel–based methods require a pre–processed input data in the form of parse trees (such as dependency analysis trees). Tree kernels have been widely applied to relation classification tasks. By analyzing trees, a contextual tree kernel has been designed by Zhou et al [10]. Other works have used tree analysis such as Bunescu and Mooney [11] who proposed to identify the relation between two entities using the SDP (Short Dependency Path).

Although these methods are effective, they still have drawbacks. Indeed, they are mainly based on statistical machine learning and their performance depends strongly on the quality of the extracted features.

Deep neural networks have recently demonstrated their capabilities to automatically learn features from data–sets. Zeng et al. [7] proposed the use of a CNN model to extract both lexical level features and sentence level features. In [12], the authors proposed a new CNN model, named CR–CNN, that offers a new pairwise loss function which allows to easily reduce the impact of artificial classes (for example, the class labeled "Other" in "SemEval 2010 task 8" dataset [13]). The CR–CNN is more efficient than the CNN model proposed by [7]. Indeed, this model uses a pairwise classification method to deal with artificial classes by omitting their embeddings, while [7] treats all of the classes equally. Compared to the CNN model [7], CR-CNN improves the F1 score by 1.4% and achieved 84.1%. Pengda Qin et al. [4] proposed a more efficient CNN for the relation classification using the Entity Tag Feature (ETF) function and achieves a better score of 84.8%. The ETF function allows to indicate the position information of the annotated entity, which is simpler but more efficient than the position function proposed in [7] and used in [12].

RNNs are another model of deep neural networks well suited to the processing of time series and sequences. One of the first works using RNN to classify relations was proposed by Socher et al. [14]. The main innovation of this model is the combination of matrix–vector (MV) representations with a recursive neural network. This model, known as MVRNN, is capable to learn both the meaning vectors of a word and how that word modifies the meaning of its neighbors through its matrix. The MVRNN model has reached an F1 score of 82.4%.

LSTM networks, an evolution of the RNN model, have significantly revolutionized the task of relationship classification. Xu et al. [6], among the first to use LSTMs, introduced SDP–LSTM, a new neural network developed to classify relations in a sentence. This model uses the multi–channel RNN with long term memory units as well as the shortest dependency path (SDP) between two entities. The SDP retains the most relevant information while Multi–channel LSTM networks allow efficient integration of information from heterogeneous sources on dependency paths. The results of SDP–LSTM are only 1.3% higher than the MVRNN [14] with an F1 score of 83.7%.

The classical LSTM model uses a combination of contexts that precede the current state and therefore only takes into account the past. However, knowing what may happen in the future can be very useful in regard to a given moment. This is the case, for example, in the word sequence processing task. For this reason, the bidirectional LSTM (BLSTM) was introduced by Zhang et al. [15]. Compared the SDP–LSTM [6], The BLSTM model applies additional information such as WordNet, dependency parser and named entity identifiers (NERs) to classify relations and achieves a more improved score of 84.3%.

Attention-based models have been proposed in recent years and are making a contribution for the relation classification task. Based on this work, Zhou et al. [16] designed the Att–BLSTM that automatically focuses on words that have a decisive effect on classification to capture the most important semantic information in a sentence, without using additional knowledge and NLP systems. Similarly, Zhang et al. [17] proposed an Att–BLSTM for the relation classification, with an attention layer for organizing the context information on the word level. [16] and [17] achieved 83.7% and 86.3% respectively of F1 score, which shows that the attention mechanism can still improve the efficiency of the neural network.

More recently, some works used the combination of two or more neural networks to perform the relation classification. In this endeavor, some works combine RNN and CNN to take advantage of both models. Guo et al. [18] have proposed a new Att–RCNN model. This model uses a combination of the two types of neural networks to capture features. Indeed, it uses an RNN to extract higher level contextual representations of words and a CNN to obtain sentence characteristics for the relation classification task.

Experiments show that the combination of networks with an attention mechanism is more suitable for the task of relation classification. For example, Att–RCNN [18] surpasses all previously models with a new F1 score of 86.6%. Indeed, it reaches nearly 4% higher than CNN [7] and nearly 3% higher than SDP–LSTM [6].

III. Proposed Approach

All the previous approaches aim at classifying relations according to nine predefined types (Cause–Effect, Content–Container, Entity–Destination, etc.) taking into account only the syntactic aspect of the text. These approaches neglect an important aspect which is the semantics, and in particular the context. An identified relation can be relevant or not depending on the context. Consider the following two sentences:

(1) The gas stove is in the kitchen.

(2) John is in a good mood.

If we rely on existing approaches to classifying relations, the two relations expressed by the two sentences above are classified in the same predefined type “Content–Container”. These approaches do not consider the context which depends on the nature of the entities. While the first sentence expresses a “Content–Container” relation, the second one expresses a state of mind which, apart from the syntax, has nothing to do with the c“Content–Container” relation.

The Context is highly essential since it helps to clarify the nature of the content of a document or what a relationship represents. In our approach, a context “ctx” is defined as the minimal information that provides a fairly clear idea of what the content of a document “D” represents. It is defined by a label “L” and a set of key words “Kw_i”. In [19], a context has been formalized as follows:

$$ctx = < Idc, L, Kw_i, i = 1..n > \tag{1}$$

In this section, we present an approach for classifying relations based on the type and the context. This approach involves five steps: (I) identification of different keywords, (II) extraction of context from the textual content, (III) identification of relations, (IV) annotation of relations by context, and finally, (V) classification of identified relations. Figure 1 summarizes the process of our approach.

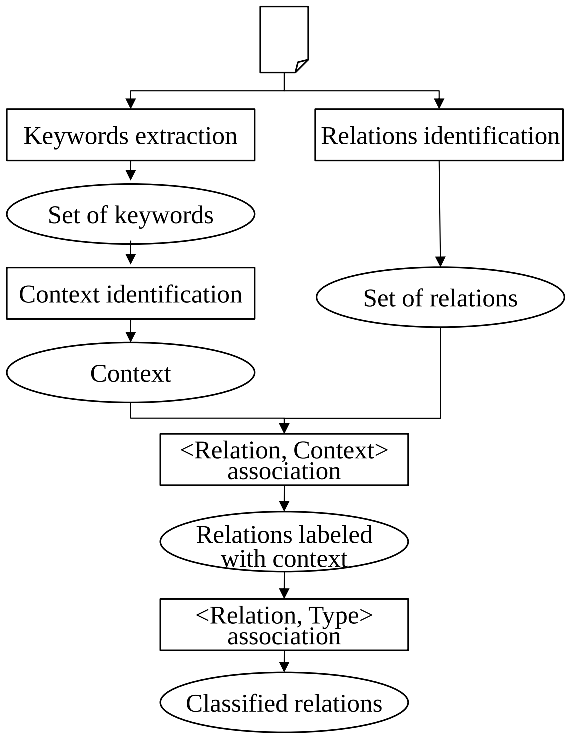

Fig. 1. Context & type based Relations classification Process

A. Keywords extraction

This task aims at identifying the relevant words that are considered as the main topics of the document. It consists of three main phases, as illustrated in Figure 2: (1) Topic extraction, (2) Candidate tag extraction, and (3) Final keyword extraction.

a) Topics extraction: More preciously, this step includes determining the most relevant words that accurately describe the content of the document. To do this, we have tested five different methods: TF–IDF [20], TextRank [21], KeyBert [22], Yake [23], and Rake [24]. Experiment results presented in section IV-B, allowed us to use the TF–IDF algorithm which achieved produced the best results when compared to the other techniques.

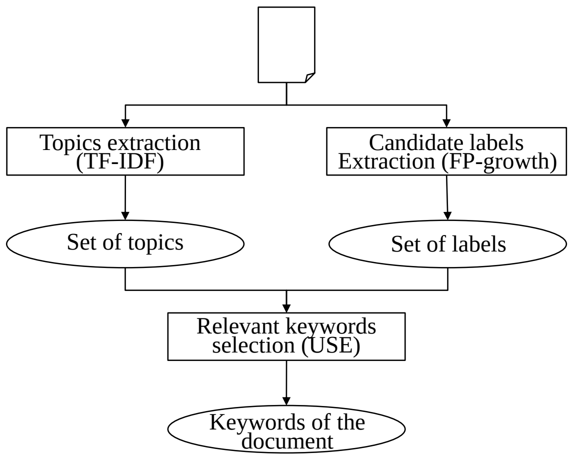

Fig. 2. Keywords Extraction process

b) Candidate labels extraction: Candidate labels are used to represent the different units that frequently appear in a document. These units are represented by unigram, bigram, or trigram words [25]. To identify these units in the document, we use the FP–growth algorithm [26].

c) Final Keywords extraction: The Universal Sentence Encoder (USE) [27] model is used, in this step, to calculate the semantic similarity between each candidate label and the various topics identified in the topics extraction stage. The USE is based on transfer learning and is trained with a deep averaging network (DAN) encoder. The final keywords of the document are defined as the five candidate labels that have the highest similarity rate with the different topics and can represent the minimal information that provides a brief overview of the text content.

B. Context Extraction

As formalized in (1), a context "ctx" is defined by a label "L" and a set of keywords "Kw_i". In this section, we focus on identifying the label "L". In order to generate the final label from the extracted keywords, we propose a LSTM model which, from the different identified keywords, allows to model a coherent sentence that represents the main idea of the document. The label generation task requires two main phases as shown in Fig 3.: (1) Training phase and (2) Modeling phase.

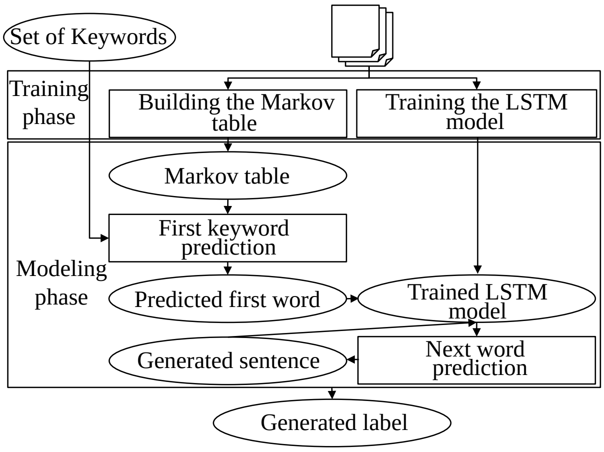

Fig. 3. Label generation process

a) Training phase: Training is the essential step in any machine learning process. It is the first step that prepares our model in order to make accurate predictions and perform the label generation task. There are two main tasks:

- **Training the LSTM model:** Starting from a set of corpora containing sentences expressed in different contexts, we trained an LSTM network. Our LSTM model consists of six layers as shown in figure 4. The embedding layer enables to convert keywords into a low dimensional dense word vectors. The two LSTM layers are separated with a dropout one to avoid over–fitting. Another dropout layer is applied after the second LSTM layer. Finally, to compute the score of each generated words predicted by the model, a "dense" layer is used with Softmax as the activation function.

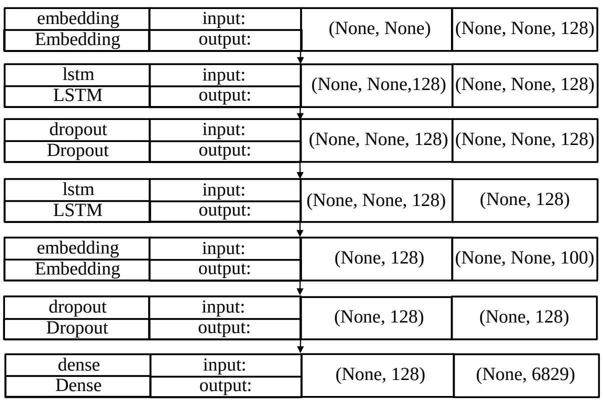

Fig. 4. Used layers of our LSTM model

- **Building the Markov table:** A Markov table is built, from meaningful corpora, through a stochastic process and which, in a bigram model for example, gives the probability that a word b is the word that follows a word a in a sentence. In our approach, the Markov table is used to decide which word should be anticipated as the first in the generated label, when the sentence ends and how the words follow each other. It is also used to check, for each iteration, the consistency of the sentence generated by our LSTM model.

b) Modeling phase: Among the 5 selected keywords, we consider the one that, according to our Markov table, has the highest probability of being the first in the sentence. Then, we use our LSTM model to predict the next word and use it as part of the prefix for the next input of the model. At this stage, we only consider the generated words that belong to the keyword list or the stop word list. The newly generated sentence is checked by the Markov table. This process is repeated until the list of extracted keywords is complete. This gives us the following algorithm:

Algorithm 1 Label modeling algorithm

function Labelmodelingtask($keywordList, stopword$)
/*Building the markov model*/
$Markovmodel \leftarrow MakovChain(keywordlist)$
$generatedlabel \leftarrow$ ""
/*prediction of the first word*/
for $word \in keywordsList$ **do**
  $firstword \leftarrow predictfirstword()$
end for
$generatedlabel \leftarrow firstword$
/*remove predicted first word from the keyword list*/
$keywordList \leftarrow removelist(firstword)$
/*Prediction of the next word*/
repeat
  $predictedword \leftarrow nextprediction(generatedlabel)$
  if ($predictedword \in keywordlist$ **or** $predictedword \in stopword$) **then**
    $generatedlabel \leftarrow genetratedlabel + predictedword$
    $keywordList \leftarrow removelist(predictedword)$
  end if
until $keywordlist = \varnothing$
return $generatedlabel$
end function

Where:

- **Markovchain():** is a method that use Markovfy library to create the table and the convertFreqIntoProb function to generate probabilities.
- **predictfirstword():** is a method that use the probabilities, given by the Markov table, of each word to be at the beginning of a sentence in order to predict the first word.
- **removelist():** is a method that aims to remove a given word from the keywordlist
- **nextprediction():** is a method that use LSTM model trained on our dataset to predict the next word.

C. Relation extraction

Stanford open–IE [28] is used in the relation extraction process. It aims to extract binary relations from a document. The system divides each sentence into a set of clauses. These latter are then maximally shortened, resulting in a collection of entailed shorter sentence fragments. The system then segments these fragments into OpenIE triples (two entities and a predicate).

D. Labeling of relations by the context

Initial relation extraction is performed with minimal extraction criteria to retrieve all relations. Consequently, the focus will be on how to select meaningful relations among all initially extracted ones. Therefore, a filtering process to eliminate the irrelevant relations is applied. The aim of this kind of filtering is to keep only those relations which are significant for the context of the document. For instance, if we consider the following paragraph (§1):

(§1) : "The 2022 French presidential election was held on 10 and 24 April 2022. Emmanuel Macron, who is married to Brigitte Trogneux, has beaten Marine Le Pen in the presidential runoff and will serve another term as president of France"

The application of the context extraction process to the paragraph (§1) allows identifying the following context: "2022 French presidential election". The relation "Emmanual Macron is married to Brigitte Trogneux" is considered as irrelevant to this context. Figure 5 illustrates the process of relations labeling by the context, which is categorized into two main stages: (1) Filtering of relations according to the context, and (2) the annotation of relations with the corresponding label.

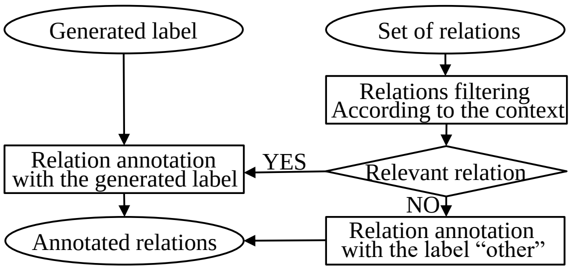

Fig. 5. The process of Labeling of relations by the context

In order to determine whether the relation is relevant or irrelevant to the context, a cosine similarity is applied to quantify the similarities between each relation and the remaining ones extracted from the document. Consequently, we must, at first, transform the relations into real–valued vectors that encode the meaning of the different words. To do this, we have evaluated three different techniques: USE [27], Bert [29] and Bert with transfer learning. Experiments, presented in section IV-B0c, allowed us to use Bert with transfer learning, which achieved the best results when compared to USE and Bert. We propose a transfer learning approach using the pre–trained model of BERT proposed by [29]. This model is capable of adjusting its parameters using a fine–tuning method in order to learn more about our WikiContext dataset and their different contexts. The vector associated with each relation is, then, compared to the vectors associated with the other relations in the document using the cosine similarity. Experiments and application of the first–order iterative optimization 'Gradient Descent' algorithm [30], allowed us to use the threshold of 0.67 as the minimum score indicating that a relation is relevant to the context.

When similarity scores are computed, two cases can be distinguished: If the relation is relevant to the context of the document, in this case, the relation is annotated with the final label of the document. Otherwise, when the relation is considered as irrelevant, we annotate it with the label "other".

E. Relation classification according to the type

The objective of this step is to classify relations annotated by context according to the type. The classification task is based on CNN model proposed by [4]. This neural network takes as input a sentence obtained in the pre–processing phase. For each input, the tokens are transformed into low–dimensional dense word vectors through Word Embedding [31]. Then, the implicit local features are extracted by the convolutional layer within multi window sizes. The most deterministic features are then captured by the "max–pooling" layer, which concatenates feature vectors from multiple windows into a vector to represent the input sentence. Following that, such sentence representations are fed into a "fully–connected" layer to compute the score of each relation.

IV. Experiments and Evaluation

To demonstrate the practical interest of our approach, a system has been developed implementing our solution. This system allows to extract and classify relations from documents according to their predefined–types and the context of documents.

A. Datasets

We use three datasets to evaluate our approach:

- Our "WikiContext" dataset: our corpus is composed of 30 contexts and created from a collection of 600 English texts published in Wikipedia. To validate the corpus, each text was manually annotated with reference keywords and a label that gives an overall idea of what a document is about. This dataset is used to perform quantitative evaluation of keywords extraction process and the labeling of relations by the context process. We use F1–score metric, to evaluate the performance of this process.
- "New York Times": To perform qualitative evaluation of the context extraction process, we consider "New York Times" dataset as a reference. In this step, we present some generated label from state–of–the–art approaches and compare them with our results. Furthermore, we use ROUGE metric (R–1,R–2 and R–L) [32] to evaluate the relevance of the generated labels.
- SemEval–2010 Task 8:This collection is used to evaluate the relation classification process based on type and context of use. Figure 6 shows an example of 10 randomly selected relations from the SemEval–2010 Task 8 test set. The first column contains the identifier id

of the sentence and the second one contains the sentence annotated with two target nominals <e1> and <e2>:

1 **Avian <e1>influenza</e1> is an infectious disease of birds caused by type A strains of the influenza <e2>virus</e2>.**
2 **The <e1>ear</e1> of the African <e2>elephant</e2> is significantly larger–measuring 183 cm by 114 cm in the bush elephant.**
3 **Skype, a free software, allows a <e1>hookup</e1> of multiple computer <e2>users</e2> to join in an online conference call without incurring any telephone costs.**
4 **This <e1>thesis</e1> defines the <e2>clinical characteristics</e2> of amyloid disease.**
5 **An FTP server is an inexpensive and relatively simple to operate tool that works great for <e1>filesharing</e1> over the <e2>internet</e2>.**
6 **The <e1>song</e1> was composed for a famous Brazilian <e2>musician</e2>.**
7 **My <e1>cat</e1> has a problem with his <e2>paw</e2>.**
8 **Essentially, the <e1>blisters</e1> that appear in the mouth are caused by the <e2>herpes simplex virus</e2> type 1, HSV-1 for short.**
9 **The <e1>treaty</e1> establishes a double majority <e2>rule</e2> for Council decisions.**
10 **Therefore, nowadays China is the complex mixture of different <e1>cultures</e1> from various <e2>epochs</e2>.**

Fig. 6. 10 randomly selected relations from the SemEval–2010 Task 8 test set

B. Results

a) Keywords extraction process results: The proposed approach for keywords extraction is tested on the "WikiContext" dataset. For each document, the output extracted from our solution is evaluated against reference keywords (annotated in our dataset. To do this, we compare the results of our solution with the results of other approaches, such as TF–IDF, TextRank, Rake, Yake, and KeyBert. The results of this evaluation are shown in Table I

TABLE I
PERFORMANCE OF KEYWORDS EXTRACTION PROCESS ON WIKICONTEXT DATASET

Method used for keywords extraction	Precision(%)	Recall(%)	F1–score
TF–IDF	46.89	47.71	47.30
TextRank	15.48	20.94	17.80
Yake	27.17	27.25	27.21
Rake	37.25	37.25	37.26
KeyBert	14.80	14.86	14.83
TF–IDF+Jiang and conrath (our approach)	49.20	65.25	56.09
TF–IDF+USE (our approach)	**65.80**	**73.20**	**69.30**

We can see that our method performs significantly better than all of the baseline techniques. The best outcome for our approach is based on combining TF–IDF, FPgrowth, and USE, and it obtains 69,30%.

b) Context Extraction process results: The context extraction process is evaluated qualitatively. Table II summarized the performance in terms of ROUGE–Measure on all the text included in the "New York Times" dataset.

TABLE II
GENERATED LABEL PERFORMANCE IN TERMS OF ROUGE MEASURE ON NEW YORK TIMES DATASET

Algorithm used for generated label process	ROUGE1	ROUGE2	ROUGEL
label generated by [33]	'f':0.3, 'p':0.5, 'r':0.22	'f':0.18, 'p':0.33, 'r':0.12	'f':0.32, 'p':0.60, 'r':0.22
label generated by [19]	'f':0.48, 'p':0.75, 'r':0.35	'f':0.21, 'p':0.55, 'r':0.13	'f':0.44, 'p':0.70, 'r':0.33
label generated by our approach	**'f':0.63, 'p':0.65, 'r':0.62**	**'f':0.27, 'p':0.58, 'r':0.18**	**'f':0.52, 'p':0.55, 'r':0.50**

We can see that our approach significantly outperforms all the baseline methods. Indeed, the ROUGE1, ROUGE2 and ROUGEL values of our approach are much better than the previous solution on the New York Times dataset.

Table III compares some of the results obtained by [33] and [19] to our approach of 5 randomly selected texts from the New "York Times" dataset

TABLE III
RESULTS PROVIDED BY OUR APPROACH COMPARED WITH THE RESULTS OBTAINED BY [33] AND [19] ON "NEW YORK TIMES" DATASET

Titles provided by New York Times	label generated by [33]	label generated by [19]	label generated by our model
2015 Was Hottest Year in Historical Record, Scientists Say Record Year Heat	Record Year Heat	The Hottest Year in the Historical Record	**2015 Was the Hottest Year in Historical Record**
Apple Settles Legal Dispute With Nokia	Apple Nokia Company	Apple and Nokia Settled a Legal Dispute	**Apple Settles a Legal Dispute**
Atlantic Hurricane Season Is Expected to Be Busy	Storm Hurricane Season Forecast	Atlantic Hurricane Season	**Atlantic Hurricane Season Is Busy**
Britain Accuses Ghana Lawmakers of Visa Fraud	Visa Ghana Parliament Britain	British authorities accused Ghana's Parliament of Visa Fraud	**Britain Accuses Ghana's Parliament of Visa Fraud**
Trump Will Withdraw U-.S. From Paris Climate Agreement	Trump Agreement Paris President	The Paris Climate Accord	**Paris Climate Agreement**

Our context extraction results show that they are content–rich and easy to understand compared with results obtained by [33] and [19].

c) Results of relations labeling by the context: In order to evaluate this process, we have tested three different techniques USE, Bert and Bert with transfer learning to determine if a relation is relevant to the context of use. We perform a transfer learning approach using the pre–trained model of BERT trained on our dataset WikiContext. Table III summarized the obtained results.

TABLE IV
LABELING BY CONTEXT PERFORMANCE IN TERMS OF F1–MESURE ON OUR WIKICONTEXT DATASET

Algorithm used for labeling by context	Precision(%)	Recall(%)	F1–score
Universal Sentence Encoder	59.15	68.25	63.37
Bert	69.50	75.20	72.23
Bert with transfer learning (our appraoch)	**72.10**	**81.20**	**76.37**

According to table IV, the pre–trained BERT model trained on our dataset outperforms Universal Sentence Encoder and Bert with F1–score of 76,37%.

d) Results of the classification based on the type and the context: In this section, our approach for relation classification according to types and context was applied. Figure 7 gives an overview of our classification task achieved on the ten sentences presented in Figure 6.

1 (Cause-Effect(e2,e1)) Influenza of birds
2 (Component-Whole(e1,e2)) ear of the African elephant
3 (Member-Collection(e2,e1)) Skype allows join online conference
4 (Message-Topic(e1,e2)) clinical characteristics of amyloid disease
5 (Other) Inexpensive and simple FTP server
6 (Product-Producer(e1,e2)) Song of Brazilian musician
7 (Component-Whole(e2,e1)) Paw of cat
8 (Product-Producer(e1,e2)) Cause of the mouth blisters
9 (Cause-Effect(e1,e2)) Council decisions
10 (Entity-Origin(e1,e2)) Different cultures of China

Fig. 7. Classification according to type and context achieved on the ten sentences presented in Table 2

The first column represents the identifier id of the sentence and the second one gives the result of the classification task: the predefined–type annotated by context.

Experiments on documents required more work. Indeed it was necessary, to determine the context of the document at the same time as the classification of the relations. If we consider a document from our dataset "WikiContext" as shown in Figure 8, the five keywords extracted and the label generated which represent the context are:

Keywords: Attacks, animal, problem, public, and health

Generated label: Animal attacks and public health problem

Figure 9 shows the relations extracted from the document presented in Figure 8 after applying the Stanford open–IE system [27].

In order to eliminate invalid relations, the labeling by context process is applied. The relation "United States in 1994" is considered as irrelevant to the "Animal attacks and public health problem" context.

Finally, the different relations are annotated by the identified context as follow (Figure 10).

Animal attacks are violent, often fatal attacks caused by animals against humans, one of the most common being bites. Bites are wounds caused as a result of an animal or human attack. These attacks are a cause of human injuries and fatalities worldwide. According to the 2012 U.S. Pet Ownership Demographics Sourcebook, 56% of United States citizens owned a pet. In the United States in 1994, approximately 4.7 million people were bitten by dogs. The frequency of animal attacks varies with geographical location, as well as hormonal secretion. Gonad glands found on the anterior side of the pituitary gland secrete androgens and estrogens hormones. Animals with high levels of these hormones tend to be more aggressive, which leads to a higher frequency of attacks not only to humans but among themselves.
Animal attacks have been identified as a major public health problem. In 1997, it was estimated that up to 2 million animal bites occur each year in the United States. Injuries caused by animal attacks result in thousands of fatalities worldwide every year. All causes of death are reported to the Centers for Disease Control and Prevention each year. Medical injury codes are used to identify specific cases.

Fig. 8. Document from WikiContext dataset

1 [entity: 'Animal attacks', relation: 'are', object: 'violent']
2 [entity: 'Bites', relation: 'are', object: 'wounds caused as a result of an animal or human attack]
3 [entity: 'attacks', relation: 'are', object: 'cause of human injuries and fatalities worldwide ']
4 [entity: 'United States citizens', relation: 'owned, object: 'a pet']
5 [entity: 'people', relation: 'were bitten by', object: 'dogs']
6 [entity: 'United States', relation: 'in', object: '1994']
7 [entity: 'frequency of animal attacks', relation: 'varies with', object: 'geographical location']
8 [entity: 'Gonad glands', relation: 'found on', object: 'anterior side of the pituitary gland']
9 [entity: 'Animals', relation: 'with', object: 'high levels of these hormones']
10 [entity: 'Animals', relation: 'tend to be', object: 'more aggressive']
11 [entity: 'Animal attacks, relation: 'have been identified as', object: 'major public health problem']
12 [entity: 'Injuries', relation: 'caused by', object: 'animal attacks']
13 [entity: 'Injuries', relation: 'result in', object: 'thousands of fatalities worldwide']
14 [entity: 'causes of death', relation: 'result in', object: 'thousands of fatalities worldwide']
15 [entity: 'causes of death', relation: 'are reported to', object: 'Centers for Disease Control and Prevention']
16 [entity: 'Medical injury codes', relation: 'are used to identify', object: 'specific cases']

Fig. 9. Different relations extracted from the document

The different results presented in Figure 10 show that our approach outperforms all the other systems with adding relations contextualization. Adding precise contextual capabilities to relations could be applied in a variety decision–making applications. Relationship contextualization, in particular when dealing with search engines, improves the effectiveness of information search.

V. CONCLUSION

In information extraction, relation classification is an important task. In the era of massive data, vital information can be contained in unstructured text documents and are therefore difficult to exploit. In order to obtain a contextual

1	**Other(e1,e2) Animal attacks and public health problem**
2	**(Content-Container(e1,e2)) Animal attacks and public health problem**
3	**(Cause-Effect(e1,e2)) Animal attacks and public health problem**
4	**(Member-Collection(e2,e1)) Animal attacks and public health problem**
5	**Other(e1,e2) Animal attacks and public health problem**
6	**Other(e1,e2) other**
7	**(Instrument-Agency(e1,e2)) Animal attacks and public health problem**
8	**(Content-Container(e1,e2)) Animal attacks and public health problem**
9	**Other(e1,e2) Animal attacks and public health problem**
10	**Other(e1,e2) Animal attacks and public health problem**
11	**(Message-Topic(e1,e2)) Animal attacks and public health problem**
12	**(Cause-Effect(e2,e1)) Animal attacks and public health problem**
13	**(Cause-Effect(e1,e2)) Animal attacks and public health problem**
14	**(Cause-Effect(e1,e2)) Animal attacks and public health problem**
15	**Other(e1,e2) Animal attacks and public health problem**
16	**(Instrument-Agency(e1,e2)) Animal attacks and public health problem**

Fig. 10. Relations annotated by the identified context

classification of relations, we propose in this paper a novel approach, to realize context classification in addition of predefined–type from relation. Experiments on three corpora show that our system outperforms the state–of–the–art relation classification systems.

There is still room for improvement despite our approach's encouraging results: (i) the proposed approach only uses deep neural networks to address relation classification. In order to improve results, we will modify our neural network by adding a bi–linear tensor layer in place of the fully connected layer. (ii) similarly, we only consider English language. It is important in our opinion to generalize this work for complex languages such as Arabic, Chinese, etc.

REFERENCES

[1] N. Bach, and S. Badaskar, "A Review for the relation extraction," Literature review for language and statistics, Carnegie Mellon University, 2007

[2] R. Lima, B. Espinasse, F. Freitas, "A logic–based relational learning approach to relation extraction: The OntoILPER system," Engineering Applications of Artificial Intelligence (EAAI) Journal, Elsevier, 2019.

[3] D. Zelenko, C. Aone, and A. Richardella, "Kernel methods for relation extraction," Journal of Machine Learning Research, 2003.

[4] P. Qin, W. Xu, and J. Guo, "An empirical convolutional neural network approach for semantic relation classification," Neurocomputing, 2016.

[5] D. Zhang, and D. Wang, "Relation classification via recurrent neural network.," arXiv preprint, 2015.

[6] Y. Xu, L. Mou, G. Li, Y. Chen, H. Peng, and Z. Jin, "Classifying relations via long short term memory networks along shortest dependency paths," Conference on Empirical Methods in Natural Language Processing, Lisbon, Portugal, 2015.

[7] D. Zeng, K. Liu, S. Lai, G. Zhou, and J. Zhao, "Relation Classification via Convolutional Deep Neural Network," 25th International Conference on Computational Linguistics, Dublin, Ireland, 2014.

[8] N. Kambhatla, "Combining lexical, syntactic, and semantic features with maximum entropy models for information extraction", In Proceedings of the ACL Interactive Poster and Demonstration Sessions, pp 78–181, 2004.

[9] G. Zhou, J. Su, J. Zhang, and M. Zhang, "Exploring various knowledge in relation extraction", In Proceedings of the 43rd annual meeting of the association for computational linguistics (acl'05), pp 427–434, 2005.

[10] G. Zhou, M. Zhang, D. Ji, and Q. Zhu, "Tree kernel–based relation extraction with context–sensitive structured parse tree information", In EMNLP–CoNLL, volume 2007, pp. 728–736. Citeseer, 2007.

[11] RC. Bunescu and RJ Mooney, "A shortest path dependency kernel for relation extraction", In Proceedings of the conference on human language technology and empirical methods in natural language processing, pp 724–731, 2005.

[12] C.N. dos Santos, B. Xiang, and B. Zhou, "Classifying relations by ranking with convolutional neural networks," In Proceedings of 53rd Annual Meeting of the Association for Computational Linguistics, 2015.

[13] P. Nakov, D.O. Séaghdha, S. Pado, M. Pennacchiotti, L. Romano, and Stan Szpakowicz, "SemEval-2010 task 8: multi-way classification of semantic relations between pairs of nominals," Proceedings of the Workshop on Semantic Evaluations: Recent Achievements and Future Directions, Boulder, Colorado, 2009

[14] R. Socher, B. Huval, C. D. Manning, and A. Y. Ng, "Semantic compositionality through recursive matrix–vector spaces," in Proc. Joint Conf. Empirical Methods Natural Lang. Process. Comput. Natural Lang. Learn., 2012.

[15] S. Zhang, D. Zheng, X. Hu and M. Yang, "Bidirectional Long Short–Term Memory Networks for Relation Classification," Proceedings of the 29th Pacific Asia Conference on Language, Information and Computation, China,pp. 73–78, 2015

[16] G. Alfattni, N. Peek, and G. Nenadic, "Attention–based bidirectional long short–term memory networks for extracting temporal relationships from clinical discharge summaries", Journal of Biomedical Informatics, 2021.

[17] R. Zhang, F. Meng, Y. Zhou, and Bing Liu. Relation classification via recurrent neural network with attention and tensor layers. Big Data Mining and Analytics, 1(3):234–244, 2018

[18] X. Guo, H. Zhang, H. Yang, L. Xu, and Z. Ye, "A single attention–based combination of cnn and rnn for relation classification", IEEE Access, 2019.

[19] M. Mallek, S. Fournier, R. Guetari, B. Espinasse and W. Chaari, "An Unsupervised Approach for Precise Context Identification from Unstructured Text Documents," International Conference on Tools for Artificial Intelligence (ICTAI), 2020.

[20] G. Salton G and C. Buckley, "Term–weighing approaches in automatic text retrieval," In Information Processing and Management, 1988

[21] R. Mihalcea R and P. Tarau, "TextRank: Bringing Order into Texts," Conference on Empirical Methods in Natural Language Processing, 2004

[22] P. sharma, "Self–supervised Contextual Keyword and Keyphrase Retrieval with Self–Labelling," 2019

[23] Campos R., Mangaravite V., Pasquali A., Jorge A.M., Nunes C., and Jatowt A., "YAKE! Collection–independent Automatic Keyword Extractor," Advances in Information Retrieval. ECIR, France, pp. 806–810, 2018.

[24] S. Rose, D. Engel, N. Cramer and W. Cowley, "Automatic keyword extraction from individual documents," Text Mining: Applications and Theory, pp.1 – 20

[25] KA. Dhand, JS. Umale and PA. Kulkarni, "Context Based Text Document Sharing System Using Association Rule Mining," Annual IEEE India Conference (INDICON), Pune, India, pp. 11–13, Dec 2014.

[26] C. Borgelt, "An Implementation of the FP–growth Algorithm," the 1st international workshop on open source data mining, Chicago, Illinois, pp. 21 – 21, 2005.

[27] Y. Yang, D. Cer, A. Ahmad, M. Guo, J. Law, N. Constant, G. Abrego , S. Yuan, Ch.Tar and R. Kurzweil, "Multilingual Universal Sentence Encoder for Semantic Retrieval," July 2019.

[28] G. Angeli, M.J. Premkumar, and C.D. Manning, " Leveraging linguistic structure for open domain information extraction," the annual meeting of the Association for Computational Linguistics (ACL), Beijing, China, 2015.

[29] J. Devlin, M. Chang, K. Lee, and K. Toutanova, "BERT: Pre–training of deep bidirectional transformers for language understanding," arXiv preprint arXiv:1810.04805, 2019.

[30] L. Nguyen, "Some Novel Algorithms for Global Optimization and Relevant Subjects," Applied and Computational Mathematics, Special Issue, Some Novel Algorithms for Global Optimization and Relevant Subjects, 2017.

[31] T. He, X. Xiang, Y. Qian, K. Yu, "Recurrent neural network language model with structured word embedding for recognition," Proceedings of ICASSP, 2015.

[32] CY. Lin, "ROUGE: A Package for Automatic Evaluation of Summaries", Association for Computational Linguistics, pp. 74–81, Barcelona, Spain, 2004.

[33] A. Sajid, S. Jan and I.A. Shah, "Automatic Topic Modeling for Single Document Short Texts," International Conference on Frontiers of Information Technology (FIT), 2017.