scientific reports

Check for updates

OPEN

A joint triple extraction method by entity role attribute recognition

Xin Jing✉, Xi Han, Bobo Li, Junjun Guo & Kun Li

In recent years, joint triple extraction methods have received extensive attention because they have significantly promoted the progress of information extraction and many related downstream tasks in the field of natural language processing. However, due to the inherent complexity of language such as relation overlap, joint extraction model still faces great challenges. Most of the existing models to solve the overlapping problem adopt the strategy of constructing complex semantic shared encoding features with all types of relations, which makes the model suffer from redundancy and poor inference interpretability in the prediction process. Therefore, we propose a new model for entity role attribute recognition based on triple holistic fusion features, which can extract triples (including overlapping triples) under a limited number of relationships, and its prediction process is simple and easy explain. We adopt the strategy of low-level feature separation and high-level concept fusion. First, we use the low-level token features to perform entity and relationship prediction in parallel, then use the residual connection with attention calculation to perform feature fusion on the candidate triples in the entity-relation matrix, and finally determine the existence of triple by identifying the entity role attributes. Experimental results show that the proposed model is very effective and achieves state-of-the-art performance on the public datasets.

Entity and Relation Extraction (ERE) aims to extract conceptual objects and their interrelations from natural language texts according to sentence semantics, and form triples similar to (Subject, Relation, Object). As a key upstream task for applications such as knowledge graph construction, intelligent question answering, and public opinion analysis, it has always occupied an important position in natural language processing. Recent studies have shown that deep learning-based joint extraction methods can significantly improve the performance of ERE by effectively integrating the interaction features between entities and relations and alleviating the error propagation problem. However, the presence of complex linguistic phenomena such as SEO (Single Entity Overlap) and EPO (Entity Pair Overlap) (shown in Table 1, from TPLinker[1]) significantly and substantially increases the design complexity of joint extraction models, thus causing the problem that model construction becomes difficult to interpret. For example, the widely adopted model of the token semantic enhancement approach[2], where the tokens of each word are spliced with other features (e.g., all types of relations) to form a synthetic encoding vector, is not only difficult to understand its ideological roots, but even flawed. The errors of this idea are obvious. First, the prediction method based on all types of relations is obviously less efficient than the search and selection method when faced with a large number of relations. Second, not all tokens in a sentence are relevant to the ERE task, and invalid and redundant information will not only increase the computational burden, but even interfere with the prediction results. Furthermore, it takes span (multiple consecutive tokens) to represent a concept in most cases, and whether a single synthetic vector can correctly represent the semantics of an entity or relational concept needs to be evaluated and verified. Thus, it is particularly important to propose explainable models that can reflect the essential process of knowledge extraction, which can help to solve the above problems and truly discover the laws of knowledge triple formation. In view of this, we simplify the design of the joint extraction model in terms of two criteria, namely, following an interpretable inference process and reducing redundant predictions, and expect to achieve the current state-of-the-art extraction performance. Our novel model benefits from the following two important inspirations.

First, inspired by the good question[3] of why simply sharing encoders is detrimental to the accuracy of both independent entity and relationship extraction. The reason for this is that the two subtasks require different features to predict entity and relation types, and sharing features without semantic correlation is counterproductive. A well-designed pipeline method is also able to beat the joint method and obtain the latest SOTA score. Through an in-depth observation of this phenomenon, we argue that the reason why humans can achieve fast and accurate results in understanding the knowledge triples contained in texts is that they can identify target objects (both entities and relations) using only shallow text features (e.g., relational trigger words) and can verify

School of Computer Science and Engineering, Xi'an Technological University, Xi'an, Shaanxi, China. ✉email: jingxin@xatu.edu.cn

	Sentences	Triplets
Normal	[The United States] President [Biden] have a meet with [Elon Reeve Musk], the CEO of [Tesla]	(The United States, *president*, Biden) (Elon Reeve Musk, *the CEO of,* Tesla)
SEO	[Stephen Chow] is a famous comedian in [China], who was born in [Hong Kong]	(Stephen Chow, *national*, China) (Stephen Chow, *born in*, Hong Kong)
EPO	[Jay Chou] is the composer and singer of ["Nocturne"]	(Jay Chou, *the composer of*, Nocturne) (Jay Chou, *the singer of*, Nocturne)

Table 1. Example of the Normal, SEO and EPO overlapping patterns.

the correctness of candidate triples guided by local semantics. Luo[4] validates the conclusion that models with feature separation strategies have better performance than models with feature fusion strategies in joint extraction tasks, which is a strong support for this observation.

Second, inspired by TPLinker[1], entities and their roles (subject or object) can be predicted separately, while disordered entity pairs and their roles can be effectively identified under relational conditions. Extending this idea, we believe that the subject-object role relationship is the most basic relationship that the triple must contain, so the problem of assignment relation type between disordered entities can be transformed into the process of triple role recognition. That is, for two entities in a sentence, if the E_1 entity is the subject of the E_2 entity and vice versa, then there must be a relationship between (E_1, E_2) that makes the subject-object relationship established. On the contrary, if there is no subject-object role relationship between (E_1, E_2), then obviously the probability that (E_1, R, E_2) is a true triple is almost zero.

Based on the above considerations, this paper proposes a novel end-to-end joint extraction model in terms of the roles that entities can play under certain relations, while incorporating a logically sound human knowledge extraction process. That is, when reading a sentence, humans can first identify the objects of entities and relations by shallow attention to the text, then they can use the identified objects to pre-assemble triples according to the role attributes, and finally they can verify the correctness of candidate triples under the guidance of complete sentence semantics. The model construction is simple, reasonable and interpretable because it is fully compatible with the human cognitive process. Since it is not limited by the sequence annotation and text generation model framework, it can naturally solve the overlapping relation extraction problem. At the same time, because only a limited number of relations are used for prediction, the noise from irrelevant relations is reduced, and the performance of the model is significantly better than that of traversing the relation set. Results of experiments on two widely used datasets, NYT[5] and WebNLG[6], show that our model beats most models that use fusion of the underlying features first and then classification, achieving competitive performance. This paper contributes as follows:

- We propose a new idea of joint triple extraction based on role attribute recognition of entities. Since the holistic semantic features of the triple better capture the correlation between entity pairs and relationships, it simplifies the process of formulating complex encoding layer feature fusion, thus effectively solving the problem of difficult interpretation of model structure.
- Combining new ideas with a rational cognitive process, we present an end-to-end triple extraction framework model. The model can realize the extraction of overlapping relations.
- Extensive experiments on two public datasets demonstrate that our model achieves comparable performance compared to state-of-the-art baselines.

Related work. Since deep learning methods can automatically and efficiently extract task-related features from sentences, deep learning-based entity-relation extraction models have been receiving extensive attention. The earliest deep learning extraction models used the pipeline approach[7–9]. As the name implies, the method treats entity and relation recognition as two separate tasks, and optimizes two independent objective functions respectively. Most studies have concluded that this method has a natural drawback[10] because it is formally unable to effectively interact the information between entities and relations. The verification of this assertion came to the opposite conclusion[3], proving that the Pipeline method can also achieve excellent performance. Their experimental results clarify two things: ① It is important to learn the different contextual representations of entities and relations. ② Better performance can be obtained only by fusing entity type information in the input layer of the relation extraction model, and the interaction between the underlying features of entities and relations is not necessary. Therefore, although the underlying lexical features are shared in this paper, a feature separation strategy is used internally for entity and relationship prediction separately.

Compared with the lack of research enthusiasm for the Pipeline method, scholars have recently proposed a large number of neural network joint extraction models, and have achieved great success. For example, Zheng[11] proposed a tagging framework that converts the joint extraction task into a tagging problem, which can directly extract entities and their relations without identifying entities and relations separately. Giannis[12] uses a CRF (Conditional Random Fields) layer to model the entity identification task and models the relationship extraction task as a multi-headed selection problem, which enables the potential identification of multiple relations per entity. Wu[13] proposed a method for enriching entity information that handles the relational classification task by locating the target entity and transmitting the information using a pre-trained architecture and merging the corresponding encodings of the two entities. However, while these efforts have solved the Normal and SEO problems, they have not yet had time to consider the more complex EPO issues.

Recent academic attention has focused on the challenge of overlapping relation extraction[14]. The approaches to solve this problem fall into two broad categories, sequence-based tagging and text-based generation. Sequence-tagging models treat the extraction task as a tagging problem, requiring the design of unique tagging schemes[1,15–17]. Text generation approaches use popular encoder-decoder architectures to generate triples[18–20], similar to machine translation approaches. However, these dominant views largely employ strategies for constructing low-level interaction features under all types of relations[16,21–23], and thus all suffer from redundant predictions and difficulties in interpretation. Different from the above methods, the model proposed in this paper only identifies limited relations in sentences, and only fuses entity and relation semantic features in the prediction stage, thus reducing the complexity of feature fusion in the encoding stage.

In addition, the information interaction between entity pairs and between entities and relations has been a key solution idea for joint extraction models. Fei[24] dynamically learn the interactions between entity spans and their relation edges through a graph attention model, which achieves effective fusion of implicit features between entity pairs. Zhang[25] apply a local focusing mechanism to entity pairs and corresponding contexts to obtain richer feature representations from local contexts to complete the RE task. Zheng[26] modify the vanilla Transformer encoder with a weighted relative position attention mechanism, which can flexibly capture the semantic feature between entities. Liu[27] models the relational graphs between the entities through a dynamic aggregation graph convolution module and gradually produces the discriminative embedded features and a refined graph through the dynamic aggregation of nodes. The commonality of these methods is that they can only satisfy the information fusion between two elements in a triad, i.e., the feature interaction between entity pairs or between a single entity and a relationship. However, a triple is composed of three elements, each of which has its own semantics and characteristics, and the feature interaction between any two combinations must be the key information for triple extraction. Therefore, this paper attempts to reduce the complexity of feature interaction by replacing the combination with the whole and learning the representations that determine the establishment of the triple directly from the three constituent elements.

Methods. We believe that a correct triple, its three components must have their own implicit attributes and strong semantic correlation. The triple itself should contain characteristics such as relation pattern, subject role, and object role. According to this key idea, we design two mechanisms in the model framework. First, possible entities and relations are identified in parallel based on the underlying lexical features, and all possible candidate triples contained in the sentence are enumerated accordingly. This mechanism can reuse the entity pair information, and thus can easily perform overlapping relation identification. Secondly, the model uses residual connection and attention mechanisms to fuse triple features and perform role attribute label prediction based on them, thus ensuring fast extraction of valid triples. In the following, we will mainly introduce the novel framework model proposed in detail through three parts: entity extractor, relation extractor, and triple extractor based on role attributes recognition (as shown in Fig. 1).

Task definition. In the joint entity and relation extraction task, the goal is to identify all possible triples (Subject, Relation, Object) in a sentence. Towards this goal, we directly model the triples and design a training objective right at the triple level. This is in contrast to previous approaches[16,28] that predict objects in the case of default subject roles. Formally, given an annotated sentence $X=\{x_t\}_{t=1}^n$ and a set of triples $T = \{(S_e, r, O_e)\}$ in X, our goal is to maximize the data likelihood of all sentences in the training set, which is defined as follows:

$$\prod_{(S_e, r, O_e) \in T} p((S_e, r, O_e)|X)$$

$$= \prod_{r \in T} p(r|X) \prod_{(S_e, O_e) \in T|r} p((S_e, O_e)|X, r) \tag{1}$$

$$= \prod_{r \in T} p(r|X) \prod_{(e_i, e_j) \in E} p((e_i, e_j)|X) \prod_{l_O^S \in L} p(l_O^S|X, r, e_i, e_j)$$

$$= \prod_{r \in T} p(r|X) \prod_{e_i \in T} p(e_i|X) \prod_{e_j \in T} p(e_j|X) \prod_{l_O^S \in L} p(l_O^S|X, r, e_i, e_j)$$

where $S_e = S \rightarrow e$ and $O_e = O \rightarrow e$ represent entities that can act as subject and object. Here S is the subject role, O is the object role, e is the entity (Note that the formula adds a letter subscript to e in order to distinguish between two different arbitrary entities), and $\rightarrow$ is the property descriptor. That is, we split the usual sense of the S (subject) and O (object) symbols into two things, the entity and its role. $r \in T$ represents the relation in the triple T. $T|r$ is the set of triples with r as the relation in T. $(S_e, O_e) \in T|r$ is an entity pair in $T|r$. E is the set of all entity pairs in x. Since the existence of two entities in an entity pair is an independent event (i.e. p(AB) = p(A)p(B)), the probability of the entity pair (e_i, e_j) can continue to be decomposed as the product of the probabilities of entities e_i and e_j. L is the set of entity role labels under a relationship. l_O^S is $\{R_mF, R_mB, Na\}$, representing the forward, reverse and disordered properties of the subject-object relationship of the entity pair, respectively.

Entity extractor. We use a multi-granularity representation vector concatenated by word embedding, character embedding and POS embedding as word embedding in a sentence. The word embedding is encoded by the pre-
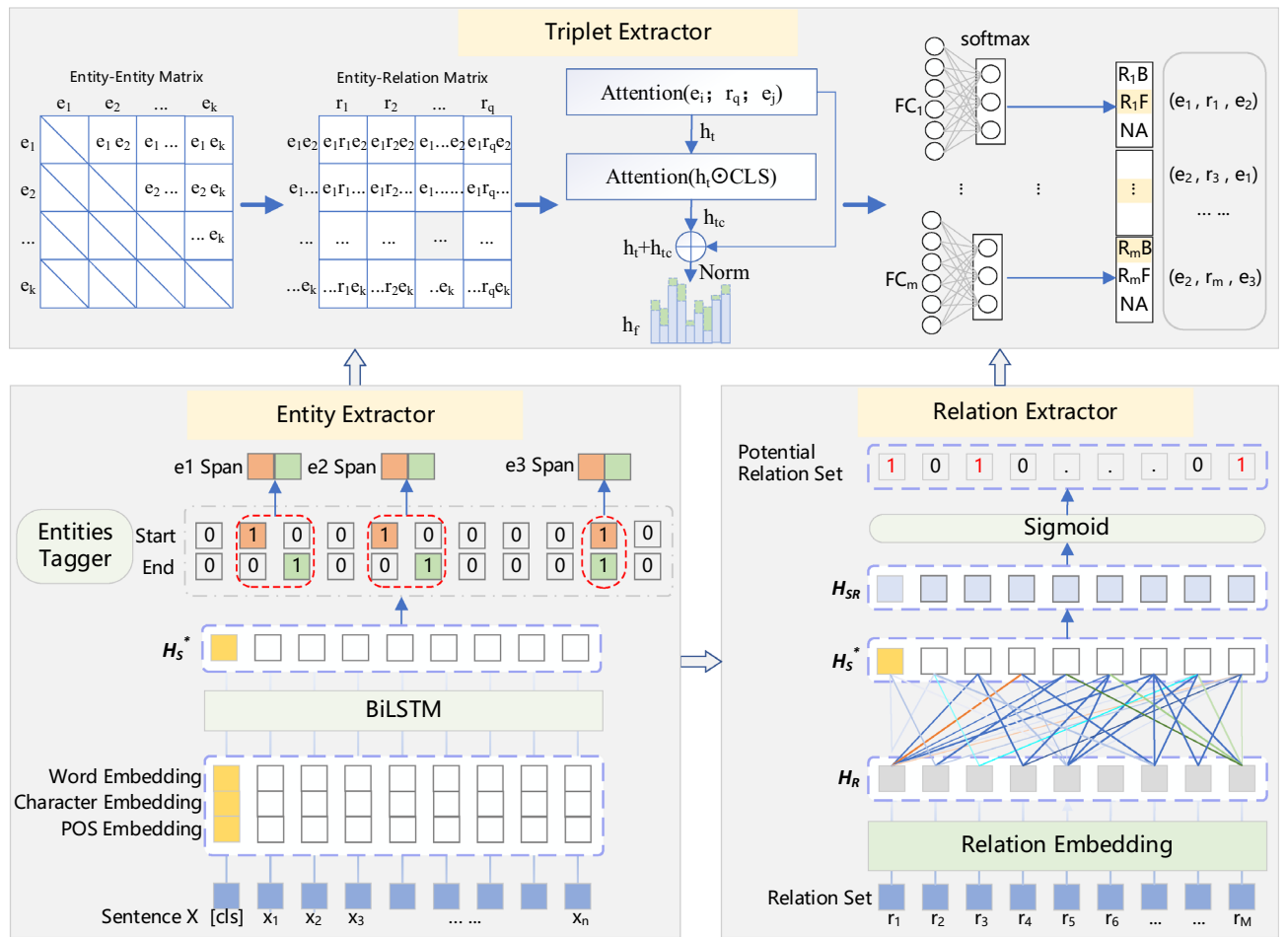
Figure 1. The framework of the proposed joint extraction model.

trained BERT[29] that captures the overall contextual features of the words. The character embedding is generated by encoding using convolutional neural networks, which effectively capture morphological information about words. Similar to other methods[21], we apply NLTK to tag sentences with POS, and randomize vector matrix of POS labels to get POS embedding. Through these methods, the embedding vector $H_S \in \mathbb{R}^{n*d}$ of a sentence can be obtained, where $d_s = d_{word} + d_{char} + d_{pos}$, and d_{word}, d_{char}, d_{pos} represent the dimension of word embedding, character embedding and POS embedding respectively. To fully integrate multi-granularity information to improve entity and relationship recognition, BiLSTM layer is used to output the final sentence representation H_S^*. We use this underlying representation for both entity and relation recognition. The process of obtaining a sentence embedding representation is as follows:

$$H_S = [h_1, h_2, \ldots, h_n] = BERT; Character; POS([x_1, x_2, \ldots, x_n]) \tag{2}$$

$$H_S^* = [h_1^*, h_2^*, \ldots, h_n^*] = BiLSTM([h_1, h_2, \ldots, h_n]) \tag{3}$$

Our entity tagger does not distinguish between header and tail entities, but directly marks the start and end positions of all possible entities in the sentence. Specifically, we use two binary classifiers to classify the last layer output of the BiLSTM directly. The detailed operations of the entity tagger on each token are as follows:

$$p_i^{start} = \sigma \left(W_{start} h_i^* + b_{start} \right) \tag{4}$$

$$p_i^{end} = \sigma \left(W_{end} h_i^* + b_{end} \right) \tag{5}$$

where p_i^{start} and p_i^{end} represent the probability of identifying the i-th token in the input sequence as the start and end position of an entity, respectively. $W_{start}, W_{end}, b_{start}, b_{end}$ are learnable parameters. $\sigma()$ is the sigmoid activation function. The corresponding token will be assigned with a tag 1 if the probability exceeds a certain threshold or with a tag 0 otherwise. We then use the proximity principle to match the start and end positions to form the entity span, and use $[h_i^*; h_j^*; width(ij)]$ as entity span representation. Where i and j are the index

numbers of the token, h_i^* and h_j^* are the feature representations of the span start and end token, respectively, and $width(ij)$ is a 20-dimensional embedding standing for the span width.

We define the training loss of entity extractor as the sum of the negative log probabilities of the true start and end tags by the predicted distributions:

$$L_{entity} = -\frac{1}{N} \sum_{i=1}^{N} \sum_{t \in \{start,end\}} \left(y_i^t \log p_i^t + (1 - y_i^t) \log(1 - p_i^t) \right) \tag{6}$$

Here, N is the length of the given sentence. y_i represents the ground truth, and p_i represents the predicted value output by the i-th token of the sigmoid.

Relation extractor. This component, shown in the bottom right corner of Fig. 1, will output a subset containing the potential relations in the sentence. Unlike previous work[22,30], the relation extractor employs the attention mechanism of the relation matrix on sentences. First, we use the method of RIFRE[16] to generate relation embedding.

$$H_R = [R_1, R_2, \ldots, R_m] = W_r E([r_1, r_2, \ldots, r_m]) + b_r \tag{7}$$

where m is the number of predefined relations, r_i is the one-hot vectors of relation indices in the predefined relations, E is the relation embedding matrix, W_r and b_r is the trainable parameters, $R_i \in \mathbb{R}^{d_h}$ is obtained by mapping the vector r_i through a linear layer after embedding. The $H_R \in \mathbb{R}^{m \times d_h}$ relation embedding matrix is used in the following steps and is updated as the relation extractor is trained. Then, we need to calculate the similarity between H_R and H_S^* to denote H_S^* by H_R. The attention-based potential relation embedding H_{SR} can be obtained as:

$$Q = H_R W_Q, K = H_S^* W_K, V = H_S^* W_V \tag{8}$$

$$H_{SR} = Softmax\left(\frac{QK^T}{\sqrt{d_k}}\right) V \tag{9}$$

where $W_Q, W_K, W_V \in \mathbb{R}^{d_h \times d_k}$ are trainable parameters, $H_{SR} \in \mathbb{R}^{m \times d_k}$, d_k is dimension of Q、K and V.

$$P_i^{rel} = \sigma\left(W_{rel} h_{SR_i} + b_{rel}\right) \tag{10}$$

Obviously, relation recognition can adopt the same multi-label classification method as entity recognition. As shown in Eq. (10), the relation tagger uses a sigmoid to obtain the probability of a potential relation after a linear feature extraction layer. If the probability exceeds a certain threshold, the corresponding token will be assigned as 1, otherwise it will be 0. Where W_{rel}, b_{rel} are learnable parameters, and $\sigma()$ is the sigmoid.

$$L_{relation} = -\frac{1}{M} \sum_{i=1}^{M} \left(y_i \log p_i^{rel} + (1 - y_i) \log(1 - p_i^{rel}) \right) \tag{11}$$

We also used cross-entropy loss for the relation extractor. Where M is the number of predefined relations, y_i represents the ground truth, and P_i^{rel} represents the predicted value output by the i-th token of the sigmoid.

Triple extractor based on role attributes recognition. The triple extractor performs entity-relation feature aggregation from a holistic perspective through residual connections, and then predicts the existence of a triple through entity role attributes. First, using the finite entities and relations predicted by the previous two modules, we can construct a finite number of candidate triples. As shown in the top left corner of Fig. 1, we enumerate all entity pairs using the entity-entity matrix. Due to the symmetry of the matrix, we only select out the set of data in the upper right corner of the matrix. Subsequently, using the entity-relation matrix we enumerate all the candidate triples. Assuming that a sentence contains K entities and Q relations, the number of candidate triples $|\varepsilon|$ can be calculated from the following equation. The minimum requirement for the presence of a candidate triple in a sentence is $K \geq 2$ and $Q \geq 1$.

$$|\varepsilon| = \frac{K(K-1)}{2} \times Q \tag{12}$$

Next, we use a residual concatenation module to obtain the triple fusion features. In the preliminary fusion layer, we regard triple as a simple graph composed of two entity nodes connected by a relation line, and fuse its features using the neighbor node information aggregation method of GAT (Graph Attention Networks)[31].

$$h_t = f\left([W_e e_i; W_r r_q; W_e e_j] + b_t \right) \tag{13}$$

where e_i, e_j represent entity span representations, $r_q \in \mathbb{R}^{d_h \times 1}$ are relation vector, W_e, W_r, b_t are trainable parameters, and f represents a fully connected layer. Feature fusion in the second layer first uses dot product attention to extract valuable information from the global sentence semantic representation [CLS] and then sums it with the h_t vector. After that, the result followed by a layer normalization operation to prevent some values from being too large, and the output representation h_f is obtained.

$$h_{tc} = Softmax\left(\frac{h_t W_{tq} * (CLS W_{tk})^T}{\sqrt{d_t}}\right) CLS \tag{14}$$

$$h_f = LayerNorm(h_t + h_{tc}) \tag{15}$$

where W_{tq}, $W_{tk} \in \mathbb{R}^{d_h \times d_t}$ are trainable parameters, h_{tc}, $h_f \in \mathbb{R}^{1 \times d_h}$. Finally, triple fusion features are input into relation-specific fully-connected networks for classification according to the relation type to determine whether there is a positive-order (Subject, Object), reverse-order (Object, Subject) or None role order attributes among the triple components, which can be formalized as:

$$p_i^{role} = softmax\left(FC_r\left(h_f\right)\right) \tag{16}$$

where p_i^{role} indicates the probabilities of role types such as R_mF, R_mB and Na (i.e., not a triple). If the predicted value is R_mF, the result will be output in the order of the input of the candidate triples, and the opposite if it is R_mB. NA can filter out low confidence triples.

$$L_{role} = -\frac{1}{M}\sum_{i=1}^{m}\sum_{j=1}^{3} y_{i,j} \log p_{i,j}^{role} = -\frac{1}{M}\sum_{i=1}^{m} \log p_i^{role} \tag{17}$$

The loss function for role attribute recognition is defined as the negative log-likelihood of the multi-classification tasks. Where y is the ground truth and (i, j) denotes the j-th label component of the i-th relation. $p_{i,j}^{role}$ denotes the probability of being predicted as the j-th role under the i-th relation. Since only one value of the j component in $y_{i,j}$ is 1, the loss function can be expressed as the sum of the corresponding probabilities of M relation truth labels. And since we use a masking mechanism for the FC fully connected layer, i.e., an all-zero mask for FC inputs with no relations, L_{role} is actually the sum of the losses of finite relations.

Training detail. From Eq. (1), it is clear that the triple prediction requires the participation of two independent entities. Therefore, we add an additional double-weight ($\alpha = 2$) for L_{entity}. Finally, the total loss is calculated as the sum of entity extraction loss, relation extraction loss, and triple extraction loss.

$$L_{total} = \alpha L_{entity} + L_{relation} + L_{role} \tag{18}$$

We minimize L_{total} and train the model in a two-step method. First, the entity extractor and the relation extractor are trained synchronously so that the model has the best initialization parameters, and then the whole model is jointly trained in an end-to-end manner.

Experiments.

Datasets. To facilitate the comparison of the triple extraction model based on role attribute recognition with other popular approaches, we evaluate our model on two public datasets NYT and WebNLG. There are two versions of these two datasets according to the annotation standard. For the fairness of the experiment, we used the most popular version from the previous work[30]. That is, we selected the datasets that are annotated with only the last word of entity. The statistics of the datasets are described in Table 2. We further described the overlapping pattern and the number of triplets per sentence in the test set.

Evaluation. We use partial matching as an evaluation metric for comprehensive experiments, i.e., an extracted triple is regarded as correct only if it is an exact match with ground truth, which means the last word of entities of both subject and object and the relation are all correct. Meanwhile, we report the standard Precision (Prec.), Recall (Rec.), and F1-score as in line with all the baselines.

Implementation details. Our model is implemented based on Tensorflow and Keras. Our BERT encoder uses the BERT-Base-Cased version, which contains 12 transformer layers and the last hidden layer output size is 768 dimensions. And the single-layer BiLSTM encoder also outputs the 768-dimensional vector. We set the maximum length of sentences to 100 and the batch size on both NYT and WebNLG datasets to 10. During model training, the learning rate of the neural network is set to 1e-5, and the Adam optimizer is used for adaptive adjustment of the weight parameters. The thresholds for both entity and relation recognition are set to 0.5. We trained the model on two RTX2080Ti graphics cards with 11G video memory. We use an early stopping strategy to prevent the model from overfitting, i.e., training is stopped if the performance on the validation set does not improve within 10 epochs.

| Dataset | Train | Valid | Test | Details of Test Set | | | Number of triplets | | | |
				Normal	SEO	EPO	N = 1	N > 1	Triples	Relations
NYT	56195	5000	5000	3266	1297	978	3244	1756	8110	24
WebNLG	5019	500	703	245	457	26	266	437	1591	171

Table 2. Statistics of datasets used in our experiments where N is the number of triples in a sentence.

Baseline methods. We compare our model with the following baselines: (1) GraphRel[32]; (2) ETL-Span[33]; (3) WDec[19]; (4) CasRel[15]; (5) TPLinker[1]; (6) CGT[34]; (7) PRGC[30]; (8) LAPREL[22]; (9) RGAM[35]; (10) BiRTE[36]. All the reported results of the baseline models are directly taken from the original literature. We refer to our model as RoleAttrTE (Triple Extraction based on Role Attributes).

Experimental results and analysis. *Main results.* The main results are shown in Table 3. Compared with other baselines on WebNLG, our proposed model achieves the same level of best results in terms of F1, and achieves almost all the best results in terms of accuracy. RoleAttrTE's F1 results at NYT were slightly worse, but still competitive. This is consistent with the recognition that the error propagation problem of the pipeline method is detrimental to accuracy, but RoleAttrTE brings more benefits in eliminating redundant predictions and facilitating frame construction. On both datasets, RoleAttrTE outperforms most methods such as the table-filling model TPLinker, and the cascade decoding model CasRel, which proves the validity of our hypothesis. This is very meaningful because it shows that: first, the method of low-level feature separation and high-level concept fusion is effective; second, entity role attributes can be identified as key labels for triple extraction.

Analysis on submodule. RoleAttrTE is essentially a two-stage model. The entity and relation extraction in the first stage will naturally affect the triple extraction in the second stage, so it is necessary to analyze the performance of these two submodules first. Table 4 gives the results of RoleAttrTE for extracting entities and relations on the two datasets. In terms of relation extraction, the model has similar F1 results on both datasets with a gap of only 0.3%, and the gap in recall is only 0.8% in the case of close precision. This indicates that the relation extraction submodule based on the attention mechanism works stably and performs well. In terms of entity extraction, RoleAttrTE achieves an impressive F1 value of 98.4% on WebNLG, while the result on NYT lags significantly by 3.1%. The reason for this is that the large size of the NYT dataset and the large number of entity types make the entity extraction submodule appear to have insufficient learning ability. This weakness will be improved when a larger parameter version of BERT is considered. Combined with the results in Table 3, we can learn that an important reason for the poor performance of RoleAttrTE on NYT is the poor entity extraction performance.

For the triple extraction submodule, we conduct a detailed set of evaluations on RoleAttrTE, which includes six cases. ① RoleAttrTEadd: additive feature fusion method. That is, the concatenate operation (;) in Eq. (13) is replaced by the addition operation (+). ② RoleAttrTE2w: dual-entity feature extraction method. That is, the two W_e in Eq. (13) are set to W_s and W_o for extracting the features of subject and object, respectively. ③ RoleAttrT-Encls: no global contextual attention feature fusion method. That is, the calculation of Eq. (14) is not performed, while Eq. (15) becomes $h_f = h_t$. ④ RoleAttrTEwhole: single network prediction method. That is, instead of using a relation-specific classification network for entity role attribute recognition, only a fully connected network is used for all relation types with prediction labels {RF, RB, Na}. ⑤ RoleAttrTEwholeM: the same as case 4 uses a single fully connected network for prediction, but the prediction labels are { R_1F, R_1B, …, R_mF, R_mB, Na}, where m represents the relation type. ⑥ RoleAttrTE: The method proposed in "Methods" section. The results in Table 5

Model	NYT			WebNLG		
	Prec	Rec	F1	Prec	Rec	F1
GraphRel	63.9	60.0	61.9	44.7	41.1	42.9
ETL-Span	85.5	71.7	78.0	84.3	82.0	83.1
WDec	94.5	76.2	84.4	-	-	-
CasRel	89.7	89.5	89.6	93.4	90.1	91.8
TPLinker	91.3	92.5	91.9	91.8	92.0	91.9
CGT	**94.7**	84.2	89.1	92.9	75.6	83.4
PRGC	93.3	91.9	92.6	94.0	92.1	93.0
LAPREL	90.7	91.4	91.1	91.7	91.5	91.6
RGAM	90.6	92	91.3	93.5	91.9	92.6
BiRTE	92.2	**93.8**	**93.0**	93.2	**94.0**	**93.6**
RoleAttrTE	91.8	91.1	91.4	**94.8**	92.4	**93.6**

Table 3. Main results, where the bolded marks are the highest scores, and the baseline result data comes from the original literature.

Task	NYT			WebNLG		
	Prec	Rec	F1	Prec	Rec	F1
Entity	94.4	96.3	95.3	98.2	98.6	98.4
Relation	95.9	96.0	95.9	95.7	96.8	96.2

Table 4. Performance of our model for two tasks of entity extraction and relation extraction on two datasets.

show that our proposed method has the best performance for the following reasons. ① The additive feature fusion method may cover part of the original valid semantic information of entities and relations. ② The dual-entity feature extraction method specifies the order of entity role attributes in advance, which is not good for predicting reverse RB type labels. ③ Obviously, CLS global semantics must contain valuable information for triple extraction. ④ It is difficult for a single prediction network to distinguish the patterns of each relationship itself, and a single prediction label cannot be used to simply replace the entity role attributes of each relationship.

Analysis on different sentence types. To verify the ability of RoleAttrTE in handling sentences with overlapping or multiple relations, we divided the two data sets according to the overlapping type and the number of triples contained in the sentences. The results of comparison with the typical baseline model are shown in Table 6. It is observed that RoleAttrTE outperforms GraphRel, CasRel, TPLinker and LAPREL in almost all tested items, and has similar performance to PRGC and BiRTE. The extraction effect of RoleAttrTE is stable and does not fluctuate greatly with the increase of sentence complexity. The results show that RoleAttrTE has advantages in dealing with overlapping problems.

Analysis on prediction errors. We classify the triples of extraction errors into the following four categories. Case 1: The triple contains only incorrect entity (at least one). Case 2: The triple contains only incorrect relation. Case 3: Cases 1 and 2 occur simultaneously. That is, both the entity and the relation contained in the triplet are incorrect. Case 4: The extraction of both entities and relations is correct, but the prediction of entity role attributes is incorrect (including misclassification as NA). The following two observations can be obtained from Fig. 2. Observation 1: The total sum of errors caused by the first stage of the model (cases 1, 2, and 3, all areas except yellow in Fig. 2) is 62% and 53.2%, respectively, both exceeding the percentage of errors in the second stage (case 4, yellow area in Fig. 2). Observation 2: When only the first stage is observed, the highest percentage of entity errors is found on NYT (cases 1, 3), while the higher percentage of relation errors is found on WebNLG (cases 2, 3), which is fully consistent with the data performance in Table 4. Thus, the following conclusions are drawn. First, it can be seen from observation 1 that the overall performance of the model can be improved by improving the accuracy of entity or relation extraction. Clearly, it is much easier to improve the two subtasks independently than to construct methods that fuse the features associated with both tasks. Therefore, there is still much room for progress in traditional pipeline methods. Second, it can be seen from observation 2 that relation features play a significantly more important role in the approach of this paper. An increase in the error rate of the previous relation classification leads to a decrease in the accuracy of the later role attribute recognition. We believe that the insufficient supply of relation information in the triple feature fusion process is the main reason for this phenomenon, and a better introduction of relation correlation features can improve the prediction performance of the model.

Task	NYT			WebNLG		
	Prec	Rec	F1	Prec	Rec	F1
RoleAttrTE$_{add}$	90.2	90.1	90.1	93.1	91.4	92.2
RoleAttrTE$_{2w}$	90.9	90.4	90.6	93.9	91.5	92.6
RoleAttrTE$_{ncls}$	90.8	90.6	90.7	92.7	91.8	92.2
RoleAttrTE$_{whole}$	51.2	50.0	50.6	36.3	20.1	25.8
RoleAttrTE$_{wholeM}$	80.6	79.5	80.0	70.2	68.5	69.3
RoleAttrTE	**91.8**	**91.1**	**91.4**	**94.8**	**92.4**	**93.6**

Table 5. Evaluation of comparison methods for feature fusion and prediction of triple extraction submodule.

Model	NYT								WebNLG							
	Normal	EPO	SEO	N=1	N=2	N=3	N=4	N≥5	Normal	EPO	SEO	N=1	N=2	N=3	N=4	N≥5
GraphRel	69.6	51.2	58.2	71.0	61.5	57.4	55.1	41.1	65.8	38.3	40.6	66.0	48.3	37.0	32.1	32.1
CasRel	87.3	92.0	91.4	88.2	90.3	91.9	94.2	83.7	89.4	94.7	92.2	89.3	90.8	94.2	92.4	90.9
TPLinker	90.1	94.0	93.4	90.0	92.8	93.1	**96.1**	90.0	87.9	95.3	92.5	88.0	90.1	94.6	93.3	91.6
LAPREL	89.0	93.1	92.6	89.0	92.5	93.3	95.2	86.8	87.0	97.1	92.5	87.2	91.2	94.6	92.9	90.2
PRGC	91.0	**94.5**	94.0	91.1	93.0	93.5	95.5	**93.0**	90.4	**95.9**	93.6	89.9	91.6	95.0	94.8	**92.8**
BiRTE	**91.4**	94.2	**94.7**	**91.5**	**93.7**	**93.9**	95.8	92.1	90.1	94.3	95.9	90.2	**92.9**	**95.7**	94.6	92.0
RoleAttrTE	89.6	93.6	93.1	89.9	91.5	93.0	95.9	89.9	**93.9**	94.7	94.6	**90.8**	91.8	95.2	**94.9**	92.6

Table 6. F1 scores on sentences with different overlapping types and different numbers of triples.
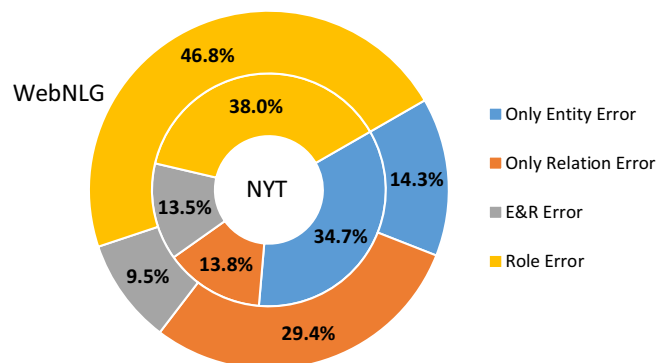
Figure 2. Comparison of triple prediction error categories.

Conclusions

This paper proposes a simple, understandable and effective ERE joint extraction model based on role attribute recognition of entities. The framework model adopts the strategy of low-level feature separation and high-level concept fusion, which naturally simulates the cognitive process of human triple extraction, and thus has strong interpretability. Meanwhile, the model can effectively solve the redundant prediction and overlapping relation problems caused by the coupling of entity and relationship features. The evaluation of the model from different aspects proves the correctness of our conjecture. ① The entity role attributes are salient features of the triple and can be identified from the holistic triple fusion features. ② In addition to their own meanings, relation features contain features of how the elements of the triple match, and can play a role in correctness discrimination in the triple fusion features. To the best of our knowledge, this paper is the first model that uses complete triple and entity role attributes as recognition objects. Experimental results show that our model achieves the same technical level as the latest SOAT on two benchmark datasets. Further analysis also demonstrated the outstanding performance of the model in handling sentences with overlapping and multiple relations.

Data availability

Our dataset access is open. Details of our dataset can be found online at https://gitee.com/JingXatu/RoleAttrTE/tree/master/Datasets.

Code availability

The model source code is available at: https://gitee.com/JingXatu/RoleAttrTE.

References

1. Wang, Y., Yu, B., Zhang, Y., Liu,T., Zhu, H., Sun, L. Tplinker: Single-stage joint extraction of entities and relations through token pair linking. In *Proceedings of the 28th International Conference on Computational Linguistics*, 1572–1582(2020).
2. Ma, L., Ren, H., Zhang, X. Effective cascade dual-decoder model for joint entity and relation extraction. arXiv e-prints. arXiv: https://arxiv.org/abs/2106.14163 (2021).
3. Zhong, Z., Chen, D. A frustratingly easy approach for joint entity and relation extraction. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, 50–61 (2021).
4. Luo, Y., Huang, Z., Zheng, K., Hao, T. Systematic analysis of joint entity and relation extraction models in identifying overlapping relations. In *Neural Computing for Advanced Applications*, 17–31(2021).
5. Riedel, S., Yao, L., Mccallum, A. Modeling relations and their mentions without labeled text. In *Joint European Conference on Machine Learning and Knowledge Discovery in Databases*, 148–163 (2010).
6. Gardent, C., Shimorina, A., Narayan, S., Perez-Beltrachini, L. Creating training corpora for nlg micro-planning. In *55th Annual Meeting of the Association for Computational Linguistics* (2017).
7. Mike, M., Steven, B., Rion, S., Dan, J. Distant supervision for relation extraction without labeled data. In *Proceedings of the 47th Annual Meeting of the ACL and the 4th IJCNLP of the AFNLP*, 1003–1011 (2009).
8. Chan, Y.S., Roth, D. Exploiting syntactico-semantic structures for relation extraction. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics*, 551–560 (2011).
9. Gormley, M.R., Yu, M., Dredze, M. Improved relation extraction with feature-rich compositional embedding models. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, 1774–1784 (2015).
10. Zhang, S. *et al.* Survey of supervised joint entity relation extraction methods. *J. Front. Comput. Sci. Technol.* **16**(04), 713–733 (2022).
11. Zheng, S., Wang, F., Bao, H., Hao, Y., Zhou, P., Xu, B. Joint extraction of entities and relations based on a novel tagging scheme. In: *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics*, 1227–1236 (2017).
12. Giannis, B., Johannes, D., Thomas, D. & Chris, D. Joint entity recognition and relation extraction as a multi-head selection problem. *Expert Syst. Appl.* **114**, 34–45 (2018).
13. Wu, S., He, Y. Enriching pre-trained language model with entity information for relation classification. In: *Proceedings of the 28th ACM International Conference on Information and Knowledge Management*, 2361–2364 (2019).
14. Nayak, T., Majumder, N., Goyal, P. & Poria, S. Deep neural approaches to relation triplets extraction: A comprehensive survey. *Cogn. Comput.* **13**(5), 1215–1232 (2021).
15. Wei, Z., Su, J., Wang, F., Tian, Y., Chang, Y. A novel cascade binary tagging framework for relational triple extraction. In: *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, 1476–1488 (2020).

16. Zhao, K., Xu, H., Cheng, Y., Li, X. & Gao, K. Representation iterative fusion based on heterogeneous graph neural network for joint entity and relation extraction. *Knowl. Based Syst.* **219**, 106888 (2021).
17. Sun, C. *et al.* Mrc4bioer: Joint extraction of biomedical entities and relations in the machine reading comprehension framework. *J. Biomed. Inform.* **125**, 103956 (2022).
18. Zeng, D., Zhang, R., Liu, Q. Copymtl: Copy mechanism for joint extraction of entities and relations with multi-task learning. In: *The Thirty-Fourth AAAI Conference on Artificial Intelligence (AAAI-20)*, 9507–9514 (2020).
19. Tapas, N., Hwee, T.N. Effective modeling of encoder-decoder architecture for joint entity and relation extraction. In: *The Thirty-Fourth AAAI Conference on Artificial Intelligence*, 8528–8535 (2020).
20. Li, X., Li, Y., Yang, J., Liu, H. & Hu, P. A relation aware embedding mechanism for relation extraction. *Appl. Intell.* **52**(9), 10022–10031 (2022).
21. Lai, T., Cheng, L., Wang, D., Ye, H. & Zhang, W. Rman: Relational multi-head attention neural network for joint extraction of entities and relations. *Appl. Intell.* **52**(3), 3132–3142 (2021).
22. Li, X., Yang, J., Hu, P. & Liu, H. Laprel a label-aware parallel network for relation extraction. *Symmetry* **13**, 961 (2021).
23. Xu, W., Yin, S., Zhao, J., Pu, T. Deep semantic fusion representation based on special mechanism of information transmission for joint entity-relation extraction. In *Pricai 2021: Trends in Artificial Intelligence*, 73–85 (2021).
24. Fei, H., Ren, Y. & Ji, D. Boundaries and edges rethinking: An end-to-end neural model for overlapping entity relation extraction. *Inf. Process. Manag.* **57**(6), 102311 (2020).
25. Zhang, H., Zhang, G. & Ma, R. Syntax-informed self-attention network for span-based joint entity and relation extraction. *Appl. Sci.* **11**(4), 1480 (2021).
26. Zheng, W., Wang, Z., Yao, Q. & Li, X. Wrtre: Weighted relative position transformer for joint entity and relation extraction. *Neurocomputing* **459**, 315–326 (2021).
27. Liu, X., Cheng, J. & Zhang, Q. Multi-stream semantics-guided dynamic aggregation graph convolution networks to extract overlapping relations. *IEEE Access* **9**, 41861–41875 (2021).
28. Zhu, M., Xue, J., Zhou, G. Joint extraction of entity and relation based on pre-trained language model. In *2020 12th International Conference on Intelligent Human-Machine Systems and Cybernetics (IHMSC)*, 179–183 (2020).
29. Devlin, J., Chang, M.-W., Lee, K., Toutanova, K. Bert: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of NAACL-HLT*, 4171–4186 (2019).
30. Zheng, H., Wen, R., Chen, X., Yang, Y., Zhang, Y., Zhang, Z. Prgc potential relation and global correspondence based joint relational triple extraction. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing*, 6225–6235 (2021).
31. Zhang, Z., Ji, H. Abstract meaning representation guided graph encoding and decoding for joint information extraction. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, 39–49 (2021).
32. Fu, T.-J., Li, P.-H., Ma, W.-Y. Graphrel modeling text as relational graphs for joint entity and relation extraction. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, 1409–1418 (2019).
33. Yu, B., Zhang, Z., Shu, X., Liu, T. Joint extraction of entities and relations based on a novel decomposition strategy. In *24th European Conference on Artificial Intelligence—ECAI 2020*, 1–8 (2020).
34. Ye, H., Zhang, N., Deng, S., Chen, M., Tan, C., Huang, F., Chen, H. Contrastive triple extraction with generative transformer. In *The Thirty-Fifth AAAI Conference on Artificial Intelligence*, 14257–14265 (2021).
35. Yang, Y., Li, X., Li, X. A relation-guided attention mechanism for relational triple extraction. In *2021 International Joint Conference on Neural Networks (IJCNN)*, 1–8 (2021).
36. Ren, F., Zhang, L., Zhao, X., Yin, S., Liu, S., Li, B. A simple but effective bidirectional framework for relational triple extraction. In *Proceedings of the Fifteenth ACM International Conference on Web Search and Data Mining*, 824–832 (2022).

Acknowledgements

Author contributions

X.J. presented the thesis idea and wrote the full paper. X.H. implemented the model and provided the experimental data. J.G. proposed the idea of algorithm improvement. B.L. and K.L. performed the figures creation and text proofreading. All authors reviewed the manuscript.

Competing interests

The authors declare no competing interests.

Additional information

Correspondence and requests for materials should be addressed to X.J.

Reprints and permissions information is available at www.nature.com/reprints.

Publisher's note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.