Computers, Materials & Continua CMC, vol.64, no.2, pp.1297-1315, 2020

A Hybrid Method of Coreference Resolution in Information Security

Yongjin Hu[1], Yuanbo Guo[1], Junxiu Liu[2] and Han Zhang[3, *]

Abstract: In the field of information security, a gap exists in the study of coreference resolution of entities. A hybrid method is proposed to solve the problem of coreference resolution in information security. The work consists of two parts: the first extracts all candidates (including noun phrases, pronouns, entities, and nested phrases) from a given document and classifies them; the second is coreference resolution of the selected candidates. In the first part, a method combining rules with a deep learning model (Dictionary BiLSTM-Attention-CRF, or DBAC) is proposed to extract all candidates in the text and classify them. In the DBAC model, the domain dictionary matching mechanism is introduced, and new features of words and their contexts are obtained according to the domain dictionary. In this way, full use can be made of the entities and entity-type information contained in the domain dictionary, which can help solve the recognition problem of both rare and long entities. In the second part, candidates are divided into pronoun candidates and noun phrase candidates according to the part of speech, and the coreference resolution of pronoun candidates is solved by making rules and coreference resolution of noun phrase candidates by machine learning. Finally, a dataset is created with which to evaluate our methods using information security data. The experimental results show that the proposed model exhibits better performance than the other baseline models.

Keywords: Coreference resolution, hybrid method, rules, BiLSTM-Attention-CRF, information security.

1 Introduction

The coreference resolution (CR) of entities is a problem in which entities are repeatedly referenced in documents, and is also the kernel in natural language processing (NLP) research [Sukthanker, Poria, Cambria et al. (2018)]. CR is primarily used to improve the performance of other NLP tasks, such as machine translation [Vaswani, Bengio, Brevdo

[1] Information Engineering University, Zhengzhou, 450000, China.

[2] Intelligent Systems Research Centre, School of Computing, Engineering & Intelligent Systems, Ulster University, Magee Campus, Northern Ireland, BT487JL, UK.

[3] Zheng Zhou University, Zhengzhou, 450001, China.

[*] Corresponding Author: Han Zhang. Email: zhang_han@zzu.edu.cn.

Received: 02 April 2020; Accepted: 21 April 2020.

et al. (2018); Lample, Ott, Conneau et al. (2018); Chen, Firat, Bapna et al. (2018)], sentiment analysis [Cambria, Poria, Hazarika et al. (2018); Etter, Colleoni, Illia et al. (2018); Ma, Peng and Cambria (2018)], relationship extraction [Zeng, Dai, Li et al. (2018); Gábor, Buscaldi, Schumann et al. (2018); Qin, Xu, and Wang (2018)], and automatic abstract generation [Liu, Flanigan, Thomson et al. (2018); Chen and Bansal (2018)]. This type of research is currently focused on the general domain, mainly because (1) there is considerable experience in CR research in the general domain, and (2) there are sufficient annotations in the general field, e.g., ACE [Doddington, Mitchell, Przybocki et al. (2004)], CoNLL-2012 [Pradhan, Moschitti, Xue et al. (2012)], and Parcor [Guillou, Hardmeier, Smith et al. (2014)]. However, there is still a gap in the field terms of research on information security.

The absence of research studies is not an indicator that research on information security is not needed. For example, "as the world's first cyber 'super destructive weapon,' Stuxnet has infected more than 45,000 networks around the World. Computer security experts believe the virus is the highest level 'worm' ever. The new virus uses a variety of advanced technologies, so it is 'extremely stealthy and destructive'." In the aforementioned sentence, the terms "Stuxnet," "the virus," "the new virus," and "it" all refer to the same entity, i.e., "Stunxnet." Using CR, we can infer that the relationship between "Stunxnet" and "the highest level worm" is "is-a," which will improve the accuracy of the relationship between the entities' attributes that are extracted from texts, thereby increasing the accuracy of the knowledge graph in information security and, in turn, threat warnings.

There are three types of CR technologies: (1) rules-based methods [Hobbs (1978); Brennan, Friedman and Pollard (1987); Lappin and Leass (1994); Lee, Chang, Peirsman et al. (2013)], (2) statistics-based methods [Soon, Ng and Lim (2001); Aone and Bennett (1995); Lee, Surdeanu and Jurafsky (2017); Wei, Yikun, Yaqian et al. (2003)], and (3) deep-learning-based methods [Wiseman, Rush, Shieber et al. (2015); Lee, He, Lewis et al. (2017); Zhang, Santos, Yasunaga et al. (2018); Wiseman, Rush and Shieber (2016); Clark and Manning (2016)]. Among these methods, (1) relies on handcrafted rules, has narrow coverage and poor flexibility, and cannot handle rich vocabulary information well. Method (3) is more applicable to fields containing large-scale annotations that can be used for CR, but is not suitable for information security that does not contain such annotations. Method (2) can handle rich lexical features, but some scholars consider (2) to be weaker than (1) in terms of accuracy [Haghighi and Klein (2009)]. Therefore, in [Lee, Surdeanu and Jurafsky (2017)], a method combining rules with statistics was used to solve the CR problem [Lee, Surdeanu and Jurafsky (2017)]. Although the method proposed in Lee et al. [Lee, Surdeanu and Jurafsky (2017)] has been used to achieve relatively ideal results in the general field, CR cannot be applied to the field of information security, which has its own particularities. The differences between information security and the general field are summarized below.

(1) Different entity types result in the extraction of different types of candidates. In the general field, entity types such as "name," "place," and "organization" are extracted, whereas the relevant entity types in information security are "product," "vulnerability," and "attack"; thus, entities usually appear in phrases such as, "advanced persistent threat."

Moreover, these entities are usually objects, and phrases such as "damage of the virus" often appear in the text, where "the virus" is also the term that must be extracted; thus, the text in information security includes noun phrases and nested phrases in noun phrases that need to be extracted, in addition to simple nouns, pronouns, and proper nouns.

(2) Different types of extracted candidates require different extraction methods. For example, the candidates that were extracted in Lee et al. [Lee, Surdeanu and Jurafsky (2017)] included all of the common nouns, proper nouns, pronouns, and syntactic patterns, such as appositive, predicate nominative, and role appositive, whereas in information security texts syntactic patterns cannot be used to extract candidates, and a different extraction method from Lee et al. [Lee, Surdeanu and Jurafsky (2017)] is required.

(3) Different features are used for CR in general and information security fields. For example, when the entity type "name" is CR-resolved in the general field, gender can be considered an important feature; whereas the entity type in information security is usually expressed in a passive voice that is gender-neutral.

(4) There is an abundance of terms, proper nouns, and abbreviations in information security text. Although abbreviations for nations or places appear in the general field, these types of abbreviations were not processed in Lee et al. [Lee, Surdeanu and Jurafsky (2017)] according to the OntoNotes annotation Guidelines [BBN Technologies (2006)].

To meet the above-mentioned challenges, a hybrid method is proposed herein to solve the CR problem in the information security field. This study was divided into two parts: (1) extracting and classifying all candidates from the text (including noun phrases, pronouns, entities, and nested phrases), and (2) applying CR to the extracted candidates. In a previous study [Han, Yuanbo and Tao (2019)], a BiLSTM+attention+CRF model was proposed to recognize named entities in documents and solved the problem of inconsistent labels of the same entity in the document, e.g., advanced persistent threat and APT. The attention mechanism was added to the BiLSTM-CRF model to focus on the relevance of a given word to all of the other words in the document: the feature representation of the word was obtained at the document level, and entity extraction and classification were then carried out. However, experiments showed that the model was slightly weak in identifying rare entities that did not appear in the training set and entities with long lengths. Inspired by the success of Lin et al. [Lin, Li and Yang (2007); Li, Savova and Kipper (2008); Wang, Zhou, Ruan et al. (2019)] in integrating domain dictionaries into the CRF and BiLSTM-CRF models to solve the problem of rare entity recognition, an improved model based on a domain dictionary (Dictionary+BiLSTM+Attention+CRF, or DBAC) was proposed. In this model, the domain dictionary matching mechanism is introduced, and new features of words and their contexts obtained according to the domain dictionary. In this way, full use can be made of the entities and entity-type information contained in the domain dictionary. The new feature is combined with the original word feature as the input of the BiLSTM+Attention+CRF model to solve the recognition problem of both rare and long entities. The candidates to be extracted include nominal phrases, pronouns, and nested phrases, in addition to entities. If only the DBAC model was used to extract nominal phrases and nested phrases, considerable manpower and material resources would be wasted to obtain annotations: noun phrases and nested phrases have certain grammatical rules that can be summarized; thus, the rules and methodology of the (DBAC) model are

adopted to extract and classify candidates from text.

The contributions of the paper are the following.

(1) A hybrid method is proposed to solve the CR problem in the information security field.

(2) A method combining rules with a deep learning model (DBAC) is proposed to solve the problem of extracting candidates in the information security field.

(3) Rules are combined with machine learning to resolve pronoun and noun phrases.

2 Related studies

CR has been studied for a long time. In the early development of CR, methods based on rules were mainly used, including the syntax-based Hobbs theory [Hobbs (1978)], centering theory based on dialogue [Brennan, Friedman and Pollard (1987)], and the syntax-based RAP algorithm [Lappin and Leass (1994)]. In the early 21st century, some scholars considered that rules-based methods perform better than machine-learning methods [Haghighi and Klein (2009)]; however, rules-based methods have clear shortcomings, namely, a high reliance on a user's ability to manually set rules because the quality of the rules directly affects the method performance, poor flexibility, and high human resources costs. Research on the use of machine learning for CR is mainly focused on training classifiers [Sukthanker, Poria, Cambria et al. (2018)], among which decision trees and random forests are the most frequently used [Soon, Ng and Lim (2001); Aone and Bennett (1995); Lee, Surdeanu and Jurafsky (2017)]. The use of deep-learning models in the NLP field has resulted in the gradual application of these methods to CR tasks [Wiseman, Rush, Shieber et al. (2015); Lee, He, Lewis et al. (2017); Zhang, Santos, Yasunaga et al. (2018); Wiseman, Rush and Shieber (2016); Clark and Manning (2016)]. In Wiseman et al. [Wiseman, Rush, Shieber et al. (2015)], the first deep-learning model for CR was proposed, and pre-training was carried out on two separate subtasks (ana preference detection and antecedent sequencing) to learn different feature representations. The model also proved that obtaining global features from entity groups could help to improve CR performance. However, the premise of the study in Wiseman et al. [Wiseman, Rush, Shieber et al. (2015)] was that the entity groups have been classified in advance, whereas for our research in information security, relevant candidates must first be extracted from the text. Therefore, in our study, the global features in the entity group in Wiseman et al. [Wiseman, Rush, Shieber et al. (2015)] were converted into the global features in the document. In Lee et al. [Lee, He, Lewis et al. (2017)], candidate detection was combined with the CR task: First, CNN was used to study the features of the characters, and LSTM was used to obtain the word features; then, the feature representations of the candidates were studied using the Attention mechanism, and the antecedents corresponding to the candidates were sorted by a feed-forward neural network. The deep neural network used in this model was very large and difficult to maintain.

In addition to these research applications in the general field, studies on using CR in the biological field have also been developed, mainly because the biological field also has large annotated corpuses, e.g., MEDSTRACT [Pustejovsky, Castano, Sauri et al. (2002)] and MEDCo [Su, Yang, Hong et al. (2008)]. Typical applications include a proposed hybrid method based on learning and rules [D'Souza and Ng (2012)] with an F1 value of

60.9%, which is the most advanced in the biological field.

3 Method

A hybrid method is proposed in this paper in which rules are combined with learning to implement a CR task. The procedure consists of two parts: (a) extracting all the candidates (including nominal phrases, pronouns, entities, and nested phrases) from the documents and classifying these candidates, and (b) performing the CR task for the candidates. The model structure is shown in Fig. 1.

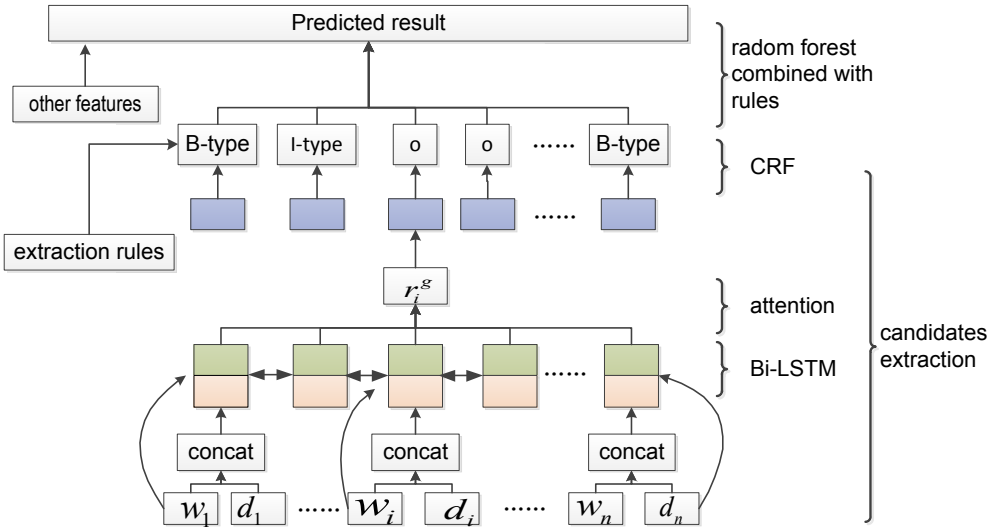

Figure 1: Model structure

In Fig. 1, r_i^g represents the feature representation of the word w_i at the document level, w_i the word embedding of the word w_i, and d_i the feature of the word w_i based on the domain dictionary. As shown in Fig. 1, the model is divided into two parts: candidate extraction and CR. Candidate extraction is a mixture of rules+DBAC, which is used to extract and classify the candidates in the text, and CR is then applied to the candidates.

3.1 Candidate extraction

In the present study, the candidate words to be extracted include nominal phrases, pronouns, entities, and nested phrases. The extraction process is divided into nominal- and nested-phrase extraction and entity extraction. The extraction of noun phrases and nested phrases adopts rules, and the extraction of entities adopts the DBAC model. The concrete architecture is shown in the candidate extraction section in Fig. 1.

3.1.1 Extraction of noun phrases and nested phrases

Normally, a noun phrase consists of a noun and its modifier, with the noun as the central word. There are two types of positional relations between modifiers and nouns: the attributive relation that is placed before a modified noun and the post-positive attributive

relation that is placed after the modified noun. An analysis of the corpus in information security shows that the noun phrases that require CR are usually prepositional attributive noun phrases. Therefore, we only consider the first positional relation.

Generally, there are two types of prepositive attributives: determiners, which are used to limit the scope of nouns, such as "these," "three," "a," "the," and "my"; and adjectives, which express the features of a noun, such as "red," "close," "new," and "small." We can obtain nominal phrases using the following rules.

We consider that U_1 represents the set of articles, U_2 the set of possessive adjectives, U_3 the set of possessive nominal pronouns, U_4 the set of demonstrative determiners, U_5 the set of quantifiers, U_6 the set of cardinal words, N the set of nouns, NP the set of noun phrases, and AD the set of adjectives; then, the set $U = U_1 \cup U_2 \cup U_3 \cup U_4 \cup U_5 \cup U_6$, from which we obtain the following rules:

(1) if $a \in U \wedge b \in N$, then $ab \in NP$;

(2) if $c \in AD \wedge b \in N$, then $cb \in NP$; and

(3) $acb \in NP$.

We summarize these three rules below as follows:

$$(\forall ab)(\forall cb)(\forall acb)((\forall a)(\forall c)(\forall b)(BEL(a,U) \vee BEL(c,AD)) \wedge BEL(b,N) \rightarrow$$
$$(\exists ab)(\exists cb)(\exists acb)BEL(ab,NP) \vee BEL(cb,NP) \vee BEL(acb,NP))$$

The term BEL refers to the predicate to which verbs belong.

In addition, we must extract the nested phrases that usually exist in the extracted noun phrases. The rules for the nested phrases to be extracted are given below.

If NNP represents the nested phrase set, ONP represents the possessive noun phrase set, and P represents the preposition set; then, the following is true.

(1) Nested phrases come from possessive noun phrases. For example, the nested phrase in the phrase "its methods" is the pronoun "its," and the nested phrase in "Stuxnet's damage" is the proper noun "Stuxnet". We summarize this rule as follows:

$$(\forall a)((\forall ab)(\forall a)BEL(ab,ONP) \wedge (BEL(a,U_2) \vee BEL(a,U_3) \rightarrow$$
$$(\exists a)BEL(a,NNP))$$

(2) Nested phrases are nouns or prepositions in nominal phrases. For example, the noun phrase "efficiency reduction" has the nested phrase "efficiency." We summarize this rule as follows:

$$(\forall a)((\forall ab)(\forall a)BEL(ab,NP) \wedge (BEL(a,N) \vee BEL(a,P)) \rightarrow$$
$$(\exists a)BEL(a,NNP))$$

If the extracted nominal phrase contains an entity, only the entity is extracted.

3.1.2 Extraction and classification of entities

1. Features based on domain dictionary

The DBAC model introduces the domain dictionary on the basis of the BiLSTM-

Attention-CRF model, and concatenates the features calculated by the domain dictionary with the word feature to form a new representation as the input of the BiLSTM-Attention-CRF model.

Consider a document $S=\{s_1,s_2,...,s_n\}$ and a domain dictionary D, where s_i is the *i*th sentence in the document, $s_i=\{w_1,w_2,...,w_m\}$; w_i is the *i*th word of the sentence; and D contains l categories of entities. n-gram is used to segment the context of the word w_i and an *l*-dimensional vector indicates whether the segmented phrase matches an entity in the dictionary. By analyzing the documents collected in the information security field, it was found that the longest entity contains five words, so here $n \le 5$ is set. Finally, a 9*l*-dimensional vector f_i is obtained for each word w_i. $f_{i,j}$ is used to represent the vector of the output corresponding to the entity type of the *j*th n-gram phrase for w_i. This feature vector $f_{i,j}$ is combined with the corresponding category representation c_j to obtain a new feature representation d_i, as shown in the Fig. 2. Here, it is assumed that *l*=6, and $c_1,c_2,...,c_6$ represents the representation of the corresponding category.

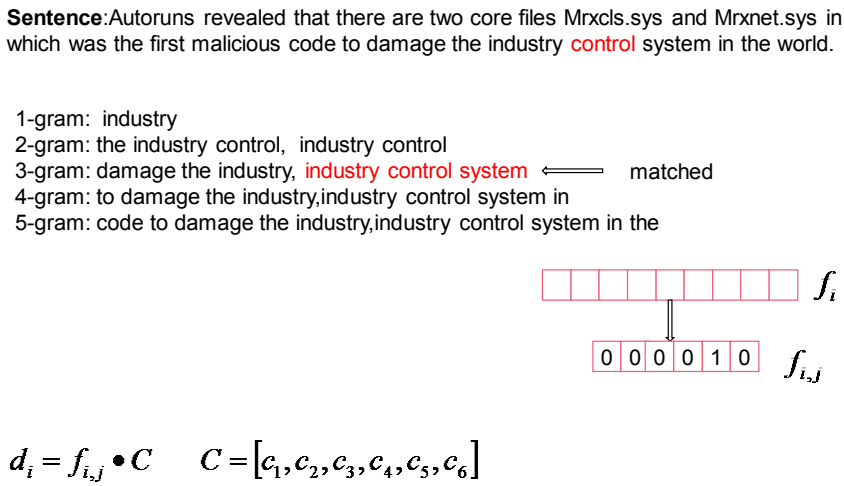

Figure 2: Example of dictionary-based feature construction

2. DBAC model

Similar to the BiLSTM-Attention-CRF model proposed in our previous work [Han, Yuanbo, and Tao (2019)], the difference here is that d_i is combined with w_i to construct a new feature w_i^{new} with the word embedding w_i as input to the model:

$$w_i^{new} = w_i \otimes d_i, \tag{1}$$

$$h_i = BiLSTM(w_i, w_i^{new}), \tag{2}$$

where h_i is the input of the Attention layer, which is mainly used to calculate the correlation degree between the words w_i and other words $w_j (j = 1,2,3,...,i-1,i+1,...,m*n)$ in the text. The weight value a_{ij} can be expressed as

$$f(w_i, w_j) = h_i^T W_a h_i, \tag{3}$$

$$a_{ij} = \frac{exp(f(w_i, w_j))}{\sum_{k=1}^{n*m} exp(f(w_i, w_k))}, \tag{4}$$

Here, W_a are the model parameters that must be trained.

A global feature representation r^g at the document level can then be obtained:

$$r_i^g = \sum_{j=1}^{N} a_{ij} h_j . \tag{5}$$

Next, a tanh layer is used to obtain the feature representation h_i^{new} of the word w_i that is related to other words in the document:

$$h_i^{new} = \tanh(W_g [r_i^g, h_i]) . \tag{6}$$

h_i^{new} is input to CRF, and the process is as follows:

$$o_i = W h_i^{new}, \tag{7}$$

$$\text{score}(D, y) = \sum_{i=1}^{N} (o_{i,y_i} + T_{y_{i-1}}, \tag{8}$$

$$y^{result} = \text{argmax}(\text{score}(D, y)) . \tag{9}$$

Here, T_{y_{i-1}, y_i} is the transform score of y_{i-1} to y_i, the function score() is used to calculate the tag sequence $y = y_1 y_2 ... y_N$ of the document D, y^{result} is the final output tag sequence result (the BIO tag), and W represents the model parameters.

3.2 CR of candidates

As there is no large-scale annotated corpus in information security for CR, a method combining rules with machine learning is proposed in this paper to carry out the CR of the candidates. The candidates for CR include pronouns and noun phrases (the entities extracted in the present study are classified as both nouns and noun phrases).

The most difficult part of the procedure is the resolution of the pronoun coreference, which is significantly related to the grammatical structure of the sentence [Sukthanker, Poria, Cambria et al. (2018)]. Therefore, this part of the study is completed using customized rules, and the noun phrase coreference is resolved using machine learning.

3.2.1 CR of pronouns

An analysis of collected texts identifies two categories of pronouns that need to be coreference resolved: relative pronouns and personal pronouns. Since characters are not entities in information security, only third-person pronouns are resolved herein.

1. CR of relative pronouns

The antecedent of a relative pronoun always appears in the same sentence and is close to its anaphora. For a relative pronoun, all of the preceding noun phrases are chosen to be its candidate antecedents. Then, according to the syntactic analysis tree of the sentence, the syntactic analysis path between the relative pronoun and the candidate word is extracted, and the shortest path is calculated. The noun phrase in the shortest path is considered to be the last antecedent of the relative pronoun. An example is given in Tab. 1.

Sentence: $(\text{Autoruns})_1$ revealed that there are $(\text{two core files})_2$ $(\text{Mrxcls.sys})_3$ and $(\text{Mrxnet.sys})_4$ in $(\text{the Stunex})_5$, $(\text{which})_7$ was (the first malicious code) to damage (the industry control system) in the World.

Table 1: Example of coreference resolution for relative pronouns

Relative pronouns	Which
Candidates	$(\text{Autoruns})_1$
	$(\text{two core files})_2$
	$(\text{Mrxcls.sys})_3$
	$(\text{Mrxnet.sys})_4$
	$(\text{the Stunex})_5$
Syntactic analysis path	NP-S-VP-SBAR-S-VP-VP-NP-SBAR-WHNP
	NP-S-VP-VP-PP-NP-SBAR-WHNP
	NP-PP-NP-SBAR-WHNP
	NP-PP-NP-SBAR-WHNP
	NP-NP-SBAR-WHNP
Shortest path	NP-NP-SBAR-WHNP (Stunex)

2. CR of third-person pronouns

The antecedent of a personal pronoun is most likely to be in the same or preceding sentence. First, candidate antecedents in the same sentence are searched for, and if the candidate set is empty, the candidate words are re-extracted from the previous sentence to find potential antecedents. Since personal pronouns must refer to entities, only security domain entity candidates are reserved. If the candidate set is not empty, the parse tree will start from the node of the personal pronoun and move upward. If there is a juxtaposed structure, including juxtaposed noun phrases, juxtaposed verb phrases, and juxtaposed clauses, the candidate word that is farthest in the first sub-structure (in terms of word distance) will be selected as the antecedent of the personal pronoun. Otherwise, the nearest clause or sentence from the parse tree is found and the furthest candidate word selected as the antecedent. An example is given in Tab. 2.

Sentence: (Stuxnet)$_1$ searches for (specific programs)$_2$, accesses (industrial control systems)$_3$, and ((its)$_?$ attack object) is the target program development tool.

Table 2: Example of coreference resolution for third-person pronouns

Third-person pronouns	its
Candidates	(Stuxnet)$_1$ (specific programs)$_2$ (industrial control systems)$_3$
Candidates for parallel structure	(Stuxnet)$_1$ searches for (specific programs)$_2$
Farthest candidate	(Stuxnet)$_1$

3.2.2 CR of noun phrases

First, the features needed for machine learning are introduced. Each feature is obtained by comparing the corresponding attributes between two items that are being resolved, as shown below.

Consistency between categories: In Section 3.1, candidates were extracted and classified. Here, whether the types of the two items being resolved are consistent is directly compared, which is a binary attribute, and the consistent case is true, and the inconsistent case is false.

Consistency between alias and abbreviation: If two items are being resolved, one is an alias or abbreviation of the other, and the value is true; otherwise, the value is false.

Consistency between singular and plural numbers: The forms of verbs or related verbs after the two items to be resolved are analyzed to determine whether the singular and plural numbers are consistent; the consistent case is true, and the inconsistent case is false.

The distance between the two items to be resolved in the text: The number of sentences between the two items to be digested in the text is confirmed.

Name similarity: For example, the phrase “the virus” usually has the same reference as the name of a virus, whereas phrases that contain words such as “product” and “company” do not.

Appositive: A syntactic analyzer is used to determine whether one of the two items to be resolved is the corresponding phrase of the other and obtain the corresponding phrase of the two items to be resolved.

Similarity of head words: in general, the head word in a noun phrase is considered to be a noun. Here, we compare the similarity of the head word in two noun phrases using cosine similarity.

Similarity of ending words: Cosine similarity is used to compare the similarity of the last word of two noun phrases.

Next, the training set is constructed. Consider that the document contains a reference chain $A_1 - A_2 - A_3 - A_4$, in which the direct adjacent reference item pairs (such as $A_1 - A_2, A_2 - A_3, A_3 - A_4$) generate a positive training sample. The extraction of

negative training samples is given below.

For example, if other objects B_1 and B_2 appear between A_1 and A_2, then a negative training sample can be derived as follows: $A_1 - B_1, A_1 - B_2, B_1 - A_2, B_2 - A_2$.

Sentence: As the World's first cyber "super destructive weapon," $(\text{Stuxnet})_{A1}$ has infected more than 45,000 networks around the World. $(\text{Computer security experts})_{B1}$ believe $(\text{the virus})_{A2}$ is $(\text{the highest level})_{B2}$ $(\text{"worm"})_{B3}$ ever. $(\text{The new virus})_{A3}$ uses a variety of advanced technologies, so it is extremely stealthy and destructive.

In the example above, the reference chain $(\text{Stuxnet})_{A1}$-$(\text{the virus})_{A2}$-$(\text{the new virus})_{A3}$ can be used to generate positive training samples: $(\text{Stuxnet})_{A1}$-$(\text{the virus})_{A2}$, $(\text{the virus})_{A2}$-$(\text{the new virus})_{A3}$. Although referential objects are transitive, we only consider short referential relationships to reduce errors. Similarly, negative training samples can be generated as follows:

$(\text{Stuxnet})_{A1}$-$(\text{Computer security experts})_{B1}$, $(\text{Computer security experts})_{B1}$-$(\text{the virus})_{A2}$,…

CR is a candidate classification problem. Therefore, we adopt the random forest algorithm. This algorithm is a classifier containing multiple decision trees that is easy to implement and has little computational overhead.

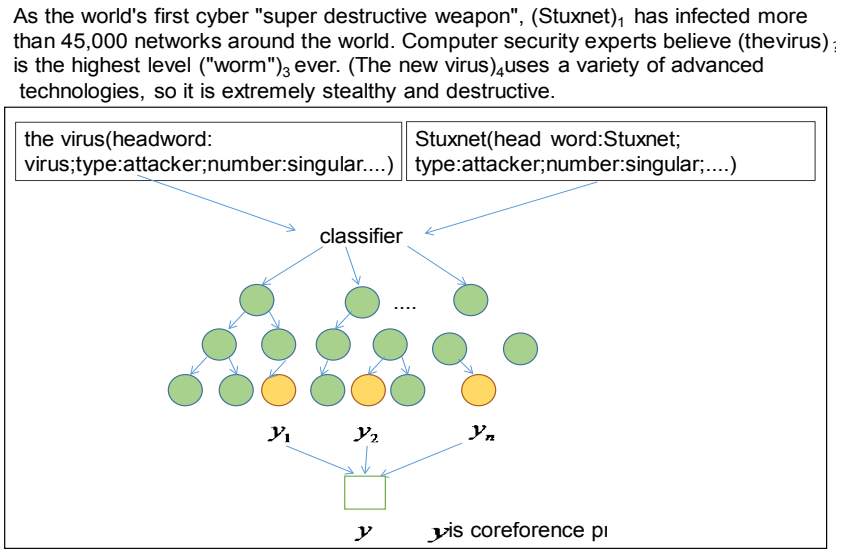

Figure 3: Examples of coreference resolution

Fig. 3 illustrates the process of using the random forest algorithm for CR. Consider that resolving the candidate word "the virus" is the goal at this time: The algorithm first links the candidate word to all of the possible antecedents within the scope of a certain sentence (in general, all of the noun phrases in two consecutive sentences are chosen). Antecedents beyond this scope are not considered. The antecedent in the antecedent chain with the highest confidence is selected as the antecedent of the candidate word. The over-generation of a referential chain is controlled by setting a minimum confidence threshold

t_i. If no confidence value is greater than t_i, the candidate word has no co-referential antecedent (this state may be changed during subsequent digestion). t_i can be obtained by training.

4 Experiments and results analysis

4.1 Data sources

Experimental data collected in our previous study [Han, Yuanbo and Tao (2019)] of texts in the information security field were used, including articles from WeLiveSecurity and Threatpost blogs, CVE (common vulnerabilities and exposures) descriptions, Microsoft security bulletins, and abstracts of journal articles in the information security field. Twenty summaries, 45 blog articles, 59 CVE descriptions, and 50 Microsoft security bulletins were extracted, resulting in a corpus of 9123 sentences. In our previous study, these texts were annotated with entity types, and these annotated corpora were used as the training data of the DBAC model. Then, 20 security reports and 20 blogs were extracted to annotate the reference chains to obtain a total of 45,932 reference chains with 7.5% positive samples. These reference chains serve as training data for machine learning. Since there are far more negative than positive training samples, to reduce the training time, the negative sample extraction method from Lee et al. [Lee, Surdeanu and Jurafsky (2017)] was adopted. First, all of the positive samples in the training dataset were used and 10% of the negative samples randomly selected for classifier training.

Then, the classifier confidence values of all of the negative samples were checked (i.e., the estimated probability), and only the fuzzy negative training samples of the first 10% reserved, i.e., the negative training samples with the highest confidence values compared with the positive training samples. These more informative negative training samples and all of the positive training samples were used to train the final classifier.

A domain dictionary constructed previously by us [Zhang, Guo and Li (2019)] using Wikipedia and the ontology of the information field, UCO, was used.

4.2 Settings

In this study, the dimension of the feature vector is set at 300, the number of nerve cells in BiLSTM at 1000, the minimum batch_size is set at 64, and the maximum number of iterations at 100. The model parameters are updated using a method from Kingma et al. [Kingma and Ba (2019)], the learning rate is set at 10^{-3}, and l_2 is set at 10^{-5}. To avoid overfitting, dropout technology was used. The dropout values of BiLSTM and the attention layer were 0.3 and 0.5, respectively. The parameter setting in the random forest essentially consists of setting the parameters of a single decision tree. The minimum confidence threshold is 30%, minimum number of leaf nodes is 5, maximum depth is the default value, and number of decision trees is 100. These parameters were obtained through 10-fold cross-validation in the training set.

The experiment was performed on a machine with two NVIDIA GTX 1080Ti graphical processing units and 64 GB of memory, and the model was trained for approximately 1 h.

4.3 Results and analysis

First, the superiority of the proposed method for CR in information security was verified. Four baselines were used: (a) the scaffolding approach proposed in Lee et al. [Lee, Surdeanu and Jurafsky (2017)]; (b) the method proposed in Soon et al. [Soon, Ng and Lim (2001)] (this method is referred to by the authors' names (Wee et al.), as the method was not named in the paper); (c) the method proposed in Zhang et al. [Zhang, Santos, Yasunaga et al. (2018)]; and (d) the method proposed in Wiseman et al. [Wiseman, Rush and Shieber (2016)]. The baselines are applied together with the proposed model to information security data, and the experimental results are shown in Tab. 3.

Table 3: Model performance comparison

Method	P	R	F
Scaffolding approach	63.8	69.9	66.7
Wee et al.	60.3	57.2	58.7
Wiseman et al.	61.2	69.5	65.1
Zhang et al.	65.4	68.3	66.8
Proposed work	70.7	74.2	72.4

As shown in Tab. 3, the proposed model outperforms the other four models for information security. An analysis of the error samples shows that the method of Wiseman et al. is mainly based on using RNN to learn the potential global representation of each entity in the entity class group, and then RNN is used to resolve these entities. This model does not use a specific clustering method, but default entity clusterings were produced. Thus, our proposed model was used to extract the candidates from the texts and then the simple K-means clustering method used to cluster the candidates. However, the experimental result is not ideal. In our analysis, in the clustering of entities, pronouns without domain features are usually clustered together. Therefore, when learning the global feature representation of these groups, domain features that undoubtedly affect the performance of the subsequent CR are difficult to learn. However, the scaffolding approach and the method of Wee et al. both address texts from the general field. Most of the features that were developed for the aforementioned models are for entities in the general field, such as "organization" and "person". Therefore, the CR performance is not sufficiently high for information security. Zhang et al. used a biaffine attention mechanism and optimized the loss function of the candidate extraction to conduct coreference resolution, which required a significant amount of annotated training data to train parameters. Therefore, this model achieved excellent performance for the conll-2012 dataset but performed poorly for the information security dataset with limited annotations.

Next, experiments were carried out on the influence of a single feature on the proposed model. The numerical numbers corresponding to eight features are shown in Tab. 4.

Table 4: Features and corresponding numbers

Feature	Number
Categories	1
Alias and abbreviation	2
Singular and plural	3
Text distance	4
Name similarity	5
Appositive	6
Similarity of head words	7
Similarity of tail words	8

The impact of a single feature on model performance is shown in Fig. 4.

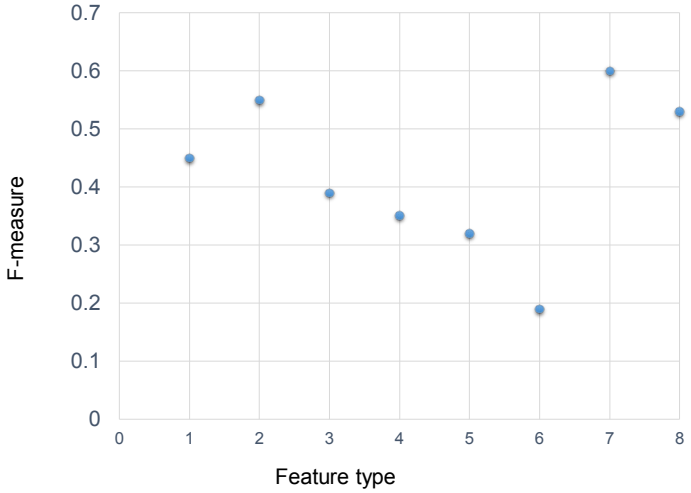

Figure 4: Impact of individual features on model performance

As shown in Fig. 4, appositive features have the least influence on CR performance of all of the features, mainly because the identification of appositive is relatively complex, such as for sentences with relatively complex grammatical structures, and the accuracy of appositive sentences that are determined only by syntactic analysis tools is not very high. In addition, the similarity of head words and the characteristics of aliases and abbreviations have the highest impact on CR performance. An analysis showed that the main reasons for these results are as follows: (a) information security text contains many professional terms and abbreviations, e.g., advanced persistent threat (APT); and (b) for noun phrases, the subject word usually determines the main meaning of the phrase.

In addition, the effect of extracting candidate words in the text also affects CR. Therefore, the experiment also verifies the performance of the proposed method for extracting candidate words (which is abbreviated as rules-based DBAC). In addition to the above-mentioned three baselines, the reference model used here also includes the method

developed by us in our previous study [Han, Yuanbo and Tao (2019)]. Here, we represent it as "previous study." The information security domain entities extracted in this experiment include the four types mentioned above: "product". "vulnerability", "attacker", and "company". The experimental results are shown in Tab. 5.

Table 5: Performance of each model for candidate extraction

Method	P	R	F
Zhang et al.	69.3	65.9	67.5
Scaffolding approach	65.1	63.8	64.4
Wee et al.	61.7	55.4	58.3
Previous study	70.1	65.2	67.6
Rules-based DBAC	76.7	82.4	79.4

As seen in Tab. 5, the proposed method (a rules-based DBAC model) outperforms the other four methods for information security, mainly because the proposed method analyzes the security text, while relying on deep learning, to summarize a set of corresponding extraction rules. This combination increases the model performance. Fig. 5 shows the entity extraction and classification results obtained using the DBAC model and our previous model (from our previous study), respectively, for the example sentence "Stunex searches for specific designed access industrial control systems".

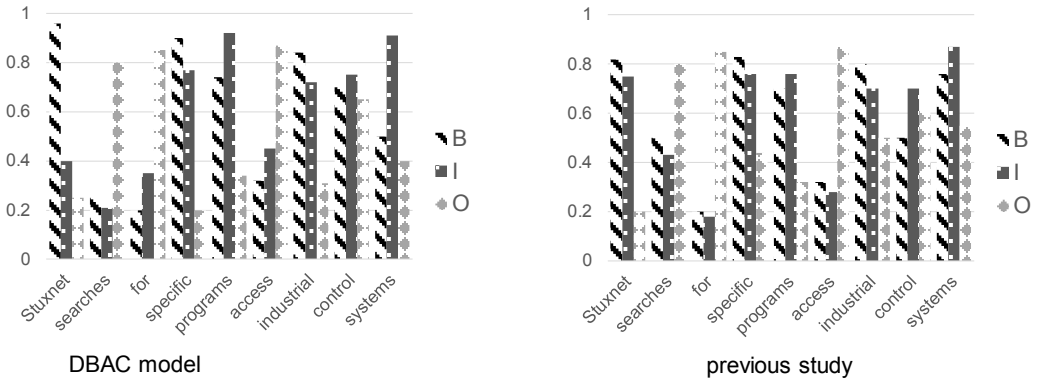

Figure 5: Extraction and classification results for DBAC model and our previous model (developed in a previous study) for the example sentence "Stunex searches for specific designed access industrial control systems"

Finally, the DBAC model is compared with the models in Li et al. [Li, Savova and Kipper (2008); Wang, Zhou, Ruan et al. (2019)] to prove the superiority of the proposed model. Since the models in the two studies were combined in different ways, the best models were selected and named using the authors' name. The information security domain entities extracted in this experiment include the four types mentioned above. The dictionary used in these models is the same one constructed in our previous work [Zhang, Guo and Li (2019)]. The comparison results are shown in Fig. 6.

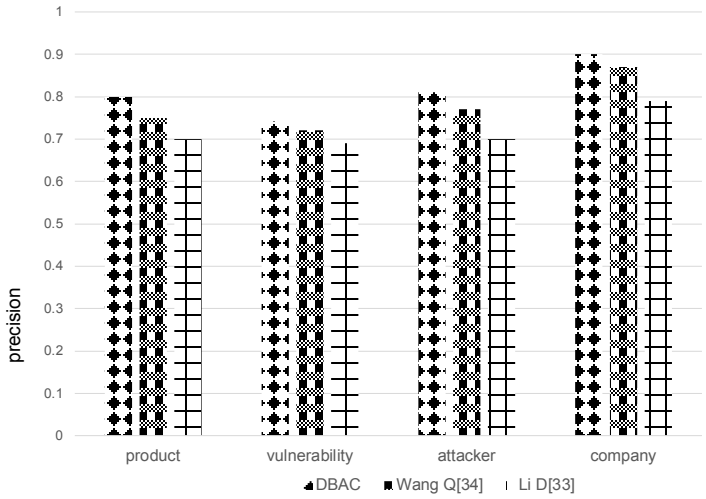

Figure 6: Comparison results of three models

Through analysis, it was found that the DBAC model could identify almost all of the long entities in the texts, while the Li D model [Li, Savova and Kipper (2008)] was weak. The feature computing method based on the domain dictionary proposed by us can make full use of the entity and entity-type information in the dictionary. In addition, since the attention mechanism was added at the document level, more word features can be captured at the document level. This is the main reason why the DBAC model performs better.

The superiority of the document-level feature in entity extraction was verified in our previous study [Han, Yuanbo and Tao (2019)] and will not be repeated here.

An analysis of the error results extracted by the rules-based DBAC model shows that the method still suffers from some problems, such as missing antecedents for words, as well as candidate words that cannot be used being obtained (i.e., candidate words with a non-coreference relationship), which still must be solved.

5 Conclusions

In this paper, a hybrid method is proposed to solve the problem of coreference resolution (CR) in the information security field. This method is mainly used to solve two problems in a CR task: (a) the extraction of all of the candidate words from the given document and the classification of those candidates, and (b) CR of the extracted candidates. A set of rules is developed according to the features of information security texts and is combined with the deep-learning model (DBAC) to solve the problem of extracting and classifying the candidate words in texts. Co-referential resolution is decomposed into pronoun co-referential resolution and noun-phrase co-referential resolution: pronoun resolution is accomplished by rules, and the co-referential resolution of noun phrases is accomplished by machine learning. The experimental results show that the proposed hybrid method is applicable to the information security field and outperforms other models that are based on the general domain. However, there are still unsolved problems in the extraction of candidates. In the future, absorbing more feature construction methods will be considered

[Li, Xu, Xian et al. (2019); Yeh (2018)] to solve these problems.

Acknowledgment: We thank anonymous reviewers for their feedback which helped in the improvement and presentation of this article. We also acknowledge Dr. Jun Ma for his feedback on an earlier version of the manuscript.

Funding Statement: This work was supported by the National Natural Science Foundation of China (grant no. 61602515).

Conflicts of Interest: We declare that there are no conflicts of interest to report regarding the present study.

References

Aone, C.; Bennett, S. W. (1995): Evaluating automated and manual acquisition of anaphora resolution strategies. *In Proceedings of the 33rd annual meeting on Association for Computational Linguistics, Association for Computational Linguistics*, pp. 122-129.

BBN Technologies. (2006): Coreference guidelines for English ontoNotes. https://www.ldc.upenn.edu/sites/www.ldc.upenn.edu/files/english-coreference-guidelines.pdf.

Brennan, S. E.; Friedman, M. W; Pollard, C. J. (1987): A centering approach to pronouns. *Proceedings of the 25th Annual Meeting on Association for Computational Linguistics*, pp. 155-162.

Cambria, E.; Poria, S.; Hazarika, D.; Kwok, K. (2018): SenticNet 5: discovering conceptual primitives for sentiment analysis by means of context embeddings. *Thirty-Second AAAI Conference on Artificial Intelligence*.

Chen, M. X.; First, O.; Bapna, A.; Johnson, M.; Macherey, W. et al. (2018): The best of both worlds: combining recent advances in neural machine translation. arXiv preprint arXiv:1804.09849.

Chen, Y. C.; Bansal, M. (2018): Fast abstractive summarization with reinforce-selected sentence rewriting. arXiv preprint arXiv:1805.11080.

Clark, K.; Manning, C. D. (2016): Deep reinforcement learning for mention-ranking coreference models. *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, pp. 2256-2262.

Doddington, G. R.; Mitchell, A.; Przybocki, M. A.; Ramshaw, L. A.; Strassel, S. et al. (2004): The automatic content extraction (ace) program-tasks, data, and evaluation. *LREC-04*, vol. 2, pp. 837-840

Kingma, D. P.; Ba, J. (2019): Adam: a method for stochastic optimization. https://arxiv.org/abs/1412.6980.

D'Souza, J.; Ng, V. (2012): Anaphora resolution in biomedical literature: a hybrid approach. *Proceedings of the ACM Conference on Bioinformatics, Computational Biology and Biomedicine*, pp. 113-122.

Etter, M.; Colleoni, E.; Illia, L.; Meggiorin, K.; D'Eugenio, A. (2018): Measuring

organizational legitimacy in social media: assessing citizens' judgments with sentiment analysis. *Business & Society*, vol. 57, no. 2, pp. 60-97.

Gábor, K.; Buscaldi, D.; Schumann, A. K.; QasemiZadeh, B.; Zargayouna, H. et al. (2018): Semeval-2018 task 7: semantic relation extraction and classification in scientific papers. *Proceedings of the 12th International Workshop on Semantic Evaluation*, pp. 679-688.

Guillou, L.; Hardmeier, C.; Smith, A.; Tiedemann, J.; Webber, B. (2014): Parcor 1.0: a parallel pronoun-coreference corpus to support statistical mt. *LREC-09*, pp. 3191-3198.

Haghighi, A.; Klein, D. (2009): Simple coreference resolution with rich syntactic and semantic features. *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, vol. 3, pp. 1152-1161.

Han, Z.; Yuanbo, G.; Tao, L. (2019): Domain named entity recognition combining GAN and BiLSTM-attention-CRF. *Journal of Computer Research and Development*, vol. 56, no. 9, pp. 1851-1858.

Hobbs, J. R. (1978): Resolving pronoun references. *Lingua*, vol. 44, no. 4, pp. 311-338.

Lample, G.; Ott, M.; Conneau, A.; Denoyer, L.; Ranzato, M. (2018): Phrase-based & neural unsupervised machine translation. arXiv preprint arXiv:1804.07755.

Lappin, S.; Leass, H. J. (1994): An algorithm for pronominal anaphora resolution. *Computational*. *Linguistics*, vol. 20, pp. 535-561.

Lee, H.; Chang, A.; Peirsman, Y.; Chambers, N.; Surdeanu, M. et al. (2013): Deterministic coreference resolution based on entity-centric, precision-ranked rules. *Computational Linguistics*, vol. 39, no. 4, pp. 885-916.

Lee, H.; Surdeanu, M.; Jurafsky, D. (2017): A scaffolding approach to coreference resolution integrating statistical and rule-based models. *Natural Language Engineering*, vol. 23, no. 5, pp. 733-762.

Lee, K.; He, L.; Lewis, M.; Zettlemoyer, L. (2017b): End-to-end neural coreference resolution. arXiv preprint arXiv:170707045

Li, D.; Savova, G.; Kipper, K. (2008): Conditional random fields and support vector machines for disorder named entity recognition in clinical texts. *Proceedings of the Workshop on Current Trends in Biomedical Natural Language Processing*, pp. 94-95.

Lin, H.; Li, Y.; Yang, Z. (2007): Incorporating dictionary features into conditional random fields for gene/protein named entity recognition. *Pacific-Asia Conference on Knowledge Discovery and Data Mining*, pp. 162-173.

Liu, F.; Flanigan, J.; Thomson, S.; Sadeh, N.; Smith, N. A. (2018): Toward abstractive summarization using semantic representations. arXiv preprint arXiv:1805.10399.

Li, Y.; Xu, G. Q.; Xian, H. Q.; Rao, L. L.; Shi, J. Q. (2019): Novel android malware detection method based on multi-dimensional hybrid features extraction and analysis. *Intelligent Automation and Soft Computing*, vol. 25, no. 3, pp. 637-647.

Ma, Y.; Peng, H.; Cambria, E. (2018): Targeted aspect-based sentiment analysis via embedding commonsense knowledge into an attentive LSTM. *Thirty-Second AAAI Conference on Artificial Intelligence*.

Pradhan, S.; Moschitti, A.; Xue, N.; Uryupina, O.; Zhang, Y. (2012): Conll-2012 shared task: modeling multilingual unrestricted coreference in ontonotes. *Joint Conference on EMNLP and CoNLL-Shared Task, Association for Computational Linguistics*, pp. 1-40.

Pustejovsky, J.; Castano, J.; Sauri, R.; Rumshinsky, A.; Zhang, J. et al. (2002): Medstract: creating large-scale information servers for biomedical libraries. *Proceedings of the ACL-02 Workshop on Natural Language Processing in The Biomedical Domain, Association for Computational Linguistics*, vol. 3, pp. 85-92.

Qin, P.; Xu, W.; Wang, W. Y. (2018): DSGAN: generative adversarial training for distant supervision relation extraction. arXiv preprint arXiv:1805.09929.

Soon, W. M.; Ng, H. T.; Lim, D. C. Y. (2001): A machine learning approach to coreference resolution of noun phrases. *Computational Linguistics*, vol. 27, no. 4, pp. 521-544.

Su, J.; Yang, X.; Hong, H.; Tateisi, Y.; Tsujii, J. (2008): Coreference resolution in biomedical texts: a machine learning approach. *Dagstuhl Seminar Proceedings, Schloss Dagstuhl-Leibniz-Zentrum fur Informatik.*

Sukthanker, R.; Poria, S.; Cambria, E. Thirunavukarasu, R. (2018): Anaphora and coreference resolution: a review. arXiv preprint arXiv:1805.11824.

Vaswani, A.; Bengio, S.; Brevdo, E.; Chollet, F.; Gomez, A. N. et al. (2018): Tensor2tensor for neural machine translation. arXiv preprint arXiv:1803.07416.

Wang, Q; Zhou, Y.; Ruan, T.; Gao, D.; Xia, Y. et al. (2019): Incorporating dictionaries into deep neural networks for the Chinese clinical named entity recognition. *Journal of Biomedical Informatics*, vol. 92, pp. 103-133.

Wei, Q.; Yikun, G.; Zhou, Y. Q.; Wu, L. (2003): English noun phrase coreference resolution via a maximum entropy model. *Journal of Computer Research and Development*, vol. 40, no. 9, pp. 1337-1343.

Wiseman, S.; Rush, A. M.; Shieber, S. (2016): Learning global features for coreference resolution. arXiv preprint arXiv:160403035.

Wiseman, S.; Rush, A. M.; Shieber, S.; Weston, J. (2015): Learning anaphoricity and antecedent ranking features for coreference resolution. *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing*, vol. 1, pp. 1416-1426.

Yeh, J. Y. (2018): Rank-order-correlation-based feature vector context transformation for learning to rank for information retrieval. *Computer Systems Science and Engineering*, vol. 33, no. 1, pp. 41-52.

Zeng, D.; Dai, Y.; Li, F.; Sherratt, R. S.; Wang, J. (2018): Adversarial learning for distant supervised relation extraction. *Computers, Materials & Continua*, vol. 55, no. 1, pp. 121-136.

Zhang, H.; Guo, Y.; Li, T. (2019): Multifeature named entity recognition in information security based on adversarial learning. *Security and Communication Networks.*

Zhang, R.; Santos, C. N.; Yasunaga, M.; Xiang, B.; Radev, D. (2018): Neural coreference resolution with deep biaffine attention by joint mention detection and mention clustering. arXiv preprint arXiv:1805.04893.