PRELIMINARY PREPRINT VERSION: DO NOT CITE
The AAAI Digital Library will contain the published
version some time after the conference.

Mastering the Explicit Opinion-role Interaction: Syntax-aided Neural Transition System for Unified Opinion Role Labeling

Shengqiong Wu,[1] Hao Fei,[1*] Fei Li,[1] Donghong Ji,[1] Meishan Zhang,[2] Yijiang Liu,[1] Chong Teng[1]

[1] Key Laboratory of Aerospace Information Security and Trusted Computing, Ministry
of Education, School of Cyber Science and Engineering, Wuhan University, China
[2] Institute of Computing and Intelligence, Harbin Institute of Technology (Shenzhen), China
{whuwsq, hao.fei, lifei_csnlp, cslyj, tengchong, dhji}@whu.edu.cn, mason.zms@gmail.com

Abstract

Unified opinion role labeling (ORL) aims to detect all possible opinion structures of 'opinion-holder-target' in one shot, given a text. The existing transition-based unified method, unfortunately, is subject to longer opinion terms and fails to solve the term overlap issue. Current top performance has been achieved by employing the span-based graph model, which however still suffers from both high model complexity and insufficient interaction among opinions and roles. In this work, we investigate a novel solution by revisiting the transition architecture, and augmenting it with a pointer network (PointNet). The framework parses out all opinion structures in linear-time complexity, meanwhile breaks through the limitation of any length of terms with PointNet. To achieve the explicit opinion-role interactions, we further propose a unified dependency-opinion graph (UDOG), co-modeling the syntactic dependency structure and the partial opinion-role structure. We then devise a relation-centered graph aggregator (RCGA) to encode the multi-relational UDOG, where the resulting high-order representations are used to promote the predictions in the vanilla transition system. Our model achieves new state-of-the-art results on the MPQA benchmark. Analyses further demonstrate the superiority of our methods on both efficacy and efficiency.[1]

1 Introduction

Interest in opinion role labeling (ORL) has been increasing in real-world applications as it fine-granularly analyzes users' opinions (as shown in Fig. 1), i.e., *what (holder) expressed what opinions towards what (target)* (Kim and Hovy 2006; Kannangara 2018; Tang et al. 2019). For ORL task, traditional work adopts pipeline methods, i.e., either first extracting all terms (opinion and role terms) and then detecting the role type (Breck et al. 2007; Yang et al. 2012) or first extracting opinion terms and then finding their corresponding role terms (Zhang, Liang, and Fu 2019; Zhang et al. 2020). Yet, such cascade schemes can result in error propagation and meanwhile ignore the interaction between opinion and role terms. Thereupon, follow-up studies design unified (aka. end-to-end) methods for ORL, i.e., modeling the task as an 'opinion-holder-target' structure extraction via LSTM-based model (Katiyar et al. 2016) or transition-based model (Zhang et al. 2019). Unified methods effectively circumvent the noise propagation and thus achieve better performances. Nevertheless, these joint models either fail to solve the term overlap issue[2], one ubiquitous case in relevant data, or stay vulnerable to the detection of lengthy mentions.

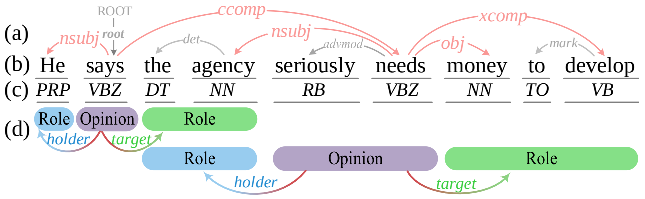

Figure 1: Illustration of the ORL structure (d) based on an example sentence (b) with its corresponding dependency structure (a) and POS tags (c).

Hopefully, a recent work (Xia et al. 2021) proposes a unified system based on the span graph model. It works by exhaustively enumerating all possible opinion and role terms, which successfully help address the term overlap challenge (Fu et al. 2021). Unfortunately, it still suffers from two crucial weaknesses. The first one is the high model complexity, owing to its span enumeration nature. Our data statistics indicate that 14.1% of opinion and role terms span more than 8 tokens, while the terms shorter than 3 tokens account for 54.8%. And the span-based graph model tries to search out as many terms as possible at the cost of huge computation and time consumption. The second issue lies in the insufficient interaction between the extraction of opinion and role terms, i.e., it takes two separate subtask decoders based on a shared encoder. Essentially, the detection of opinion and role terms should be made inter-dependent on each other. For example in Fig. 1, by capturing the interactions between opinions and roles, the recognition of 'He' as a *holder* role for the opinion 'says' could meanwhile inform the detection of 'the agency' as a *target* role for 'says', since an opinion governs both *holder* and *target*. Likewise, the role as *target*

[*] Corresponding author

Copyright © 2022, Association for the Advancement of Artificial Intelligence (www.aaai.org). All rights reserved.

[1] Codes at https://github.com/ChocoWu/SyPtrTrans-ORL

[2] For example in Fig. 1, the role term 'the agency' involves with two opinion structure simultaneously. More generally, overlap issue refers to one word participating in multiple different span terms (Zeng et al. 2018; Fei, Ren, and Ji 2020; Li et al. 2021).

of 'the agency' can further lead to the prediction of an opposite role (i.e., *holder*) for the opinion term 'seriously needs'.

In this work we try to address all the above challenges by investigating a novel unified ORL framework. First of all, we reformalize the task as structure parsing and solve it with a neural transition model. Different from Zhang et al. (2019), we redesign the transition architecture so as to support the recognition of overlapping terms. The transition system produces all possible opinion structures (including the overlapped one) in one shot by retaining the advantage of linear-time complexity. Then, we equip the transition model with a PointNet (Vinyals et al. 2015) for determining the end boundaries of opinion terms and role terms, by which we break through the restraint of lengthy terms.

We further consider the leverage of external rich syntax knowledge for enhancing the above vanilla transition system. On the one hand, we reinforce the boundary recognition of PointNet by integrating the linguistic part-of-speech (POS) tag features. Intuitively, the POS pattern could help better determine the boundary of the span terms (Lin et al. 2018). For instance, the phrase 'the agency' with the implication of 'DT-NN' POS tags is more likely to be a term. On the other hand, we propose a dependency-guided high-order interaction mechanism to achieve explicit interactions between opinions and roles. Essentially, the opinion-role structure echoes much with the dependency structure, as can be illustrated in Fig. 1, and such external syntax structure could promote the interactions between the opinion and role terms. Technically, we construct a **u**nified **d**ependency-**o**pinion **g**raph (UDOG), in which we simultaneously model the dependency structure and the partially predicted opinion-role structure as one unified graph. We especially devise a novel **r**elation-**c**entered **g**raph **a**ggregator (RCGA) to encode the multi-relational UDOG, where the resulting global-level features will be used for promoting the discovery of the rest of unknown opinion structures.

We evaluate the efficacy of our framework on the ORL benchmark, MPQA v2.0 (Wiebe et al. 2005). Experimental results demonstrate that our method obtains new state-of-the-art results than the current best-performing baseline, meanwhile achieving faster decoding. We further show that the leverage of the external syntax knowledge effectively promotes the term boundary recognition and the interactions between the detection of opinion-role structures. More in-depth analyses further reveal the strengths of our framework. We summarize the contributions of this work below:

★ We explore a novel end-to-end solution for ORL based on a neural transition model with a pointer network. The design of transition with PointNet enables more accurate predictions on longer terms, and solving the overlap issue meanwhile enjoying a linear-time complexity.

★ We strengthen the term boundary detection with linguistic POS tags. We also achieve the explicit interactions of opinion role terms by the use of syntactic dependency structure knowledge, by which we capture rich global-level high-order features and substantially enhance the predictions.

★ Our system obtains new state-of-the-art ORL results on benchmark data. Further in-depth analyses are presented for a deep understanding of our method.

2 Related Work

ORL, i.e., mining the individuals' fine-grained opinion, has been a well-established task in NLP community (Hu et al. 2004; Pang et al. 2007; Chen et al. 2016). Earlier works mostly take two-step methods for the task (Breck, Choi, and Cardie 2007; Yang and Cardie 2012; Marasović and Frank 2018; Zhang, Liang, and Fu 2019). Recent efforts consider jointly extracting the overall opinion-role results in one shot, modeling the implicit interactions between two stages meanwhile reducing the error propagation (Katiyar et al. 2016).

It is noteworthy that Zhang et al. (2019) investigate a neural transition method, achieving end-to-end task prediction while keeping a linear-time model complexity (Zhang et al. 2010; Dyer et al. 2015; Fei et al. 2021). *This work inherits the merits of the transition modeling of ORL, but ours further advances in three aspects:* (1) we re-design the transition framework so as to address a crucial *mention overlap* challenge in the task; (2) we enhance the transition model with a pointer network for the end-boundary recognition (Fernández-González et al. 2020), which significantly enables more accurate term detection, especially for those long terms. (3) we consider making use of the opinion-role interaction for better task performances.

More recently, Xia et al. (2021) solve the overlap issue with a span-based graph model by iteratively traversing all possible text spans. Unfortunately, such exhaustive enumerating task modeling results in huge computational costs, i.e., $O(n^4)$. Meanwhile, their work fails to explicitly manage the interactions between the opinion-role structures, where further task improvements can be pretty limited. Besides, Xia et al. (2021) implicitly integrate the syntactic constituency information based on a multi-task learning framework. Differently, we take the advantage of dependency syntax knowledge for building the UDOG, through which we achieve the goal of explicit and sufficient opinion-role interactions, and thus obtain substantially enhanced performances.

3 Transition System

Task Formalization. We model the task as an '$\langle$opinion, role(type)$\rangle$' structure parsing. Given a sentence X=$\{w_1,\cdots,w_T\}$, our system outputs a list of pairs Y=$\{\langle o,r(c)\rangle_q,\cdots\}_{q=1}^{Q}$, where o is an opinion term, with o=$\{w_i,\cdots,w_j|1\leq i\leq j\leq T\}$; r is a role term, with r=$\{w_{i'},\cdots,w_{j'}|1\leq i^{'}\leq j^{'}\leq T\}$; $c\in\{hd,tg\}$ is a role type, and 'hd' means '*holder*' and 'tg' represents '*target*'.

3.1 Transition Definition

Overall, the transition system comprises of actions and states, where the actions determine the parsing decision and produce transition states, while the states control the action prediction and store partial outputs.

States. The transition state is formally defined as $s = (\sigma^o,\alpha^o,\sigma^r,\alpha^r,\lambda,\beta,A,Y)$. Specifically, σ^o and σ^r are two *stacks* to store processed opinion terms and role terms, respectively. α^o and α^r are also two *stacks* for holding opinion and role terms that are popped temporarily from σ^o and σ^r, respectively. λ is a *variable* referring to either an opinion or

Sentence : He$_1$ says$_2$ the$_3$ agency$_4$ seriously$_5$ needs$_6$ money$_7$ to$_8$ develop$_9$

Step	Action	σ^o	α^o	λ	σ^r	α^r	β	Ptr	Y
0	-	[]	[]	Null	[]	[]	[1,· · · ,9]		
1	R-START	[]	[]	$(1,1)^r$	[]	[]	[$\underline{1}$,· · · ,9]	[$\underline{1}$,· · · ,9]	
2	SHIFT	[]	[]	Null	[(1,1)]	[]	[2,· · · ,9]		
3	O-START	[]	[]	$(2,2)^o$	[(1,1)]	[]	[$\underline{2}$,· · · ,9]	[$\underline{2}$,· · · ,9]	
4	ARC	[]	[]	$(2,2)^o$	[]	[(1,1)]	[2,· · · ,9]		$Y \cup \{\langle(2,2)^o, (1,1)^r(hd)\rangle\}$
5	SHIFT	[(2,2)]	[]	Null	[(1,1)]	[]	[3,· · · ,9]		
6	R-START	[(2,2)]	[]	$(3,4)^r$	[(1,1)]	[]	[$\underline{3}$,· · · ,9]	[3,$\underline{4}$,· · · ,9]	
7	ARC	[]	[(2,2)]	$(3,4)^r$	[(1,1)]	[]	[3,· · · ,9]		$Y \cup \{\langle(2,2)^o, (3,4)^r(tg)\rangle\}$
8	SHIFT	[(2,2)]	[]	Null	[(1,1),(3,4)]	[]	[4,· · · ,9]		
9	NO-START	[(2,2)]	[]	Null	[(1,1),(3,4)]	[]	[5,· · · ,9]		
10	O-START	[(2,2)]	[]	$(5,6)^o$	[(1,1),(3,4)]	[]	[$\underline{5}$,· · · ,9]	[5,$\underline{6}$,· · · ,9]	
11	ARC	[(2,2)]	[]	$(5,6)^o$	[(1,1)]	[(3,4]	[5,· · · ,9]		$Y \cup \{\langle(5,6)^o, (3,4)^r(hd)\rangle\}$
12	NO-ARC	[(2,2)]	[]	$(5,6)^o$	[]	[(1,1),(3,4]	[5,· · · ,9]		
13	SHIFT	[(2,2),(5,6)]	[]	Null	[(1,1),(3,4]	[]	[6,· · · ,9]		
14	NO-START	[(2,2),(5,6)]	[]	Null	[(1,1),(3,4]	[]	[7,8,9]		
15	R-START	[(2,2),(5,6)]	[]	$(7,9)^r$	[(1,1),(3,4]	[]	[$\underline{7}$,8,9]	[7,8,$\underline{9}$]	
16	ARC	[(2,2)]	[(5,6)]	$(7,9)^r$	[(1,1),(3,4]	[]	[7,8,9]		$Y \cup \{\langle(5,6)^o, (7,9)^r(tg)\rangle\}$
17	NO-ARC	[]	[(2,2),(5,6)]	$(7,9)^r$	[(1,1),(3,4]	[]	[7,8,9]		
18	SHIFT	[(2,2),(5,6)]	[]	Null	[(1,1),(3,4,(7,9)]	[]	[8,9]		
19	NO-START	[(2,2),(5,6)]	[]	Null	[(1,1),(3,4,((7,9)]	[]	[9]		
20	NO-START	[(2,2),(5,6)]	[]	Null	[(1,1),(3,4,(7,9)]	[]	[]		

Figure 2: An illustration of the transition process. 'Ptr' means PointNet for end boundary detection of opinion or role terms. The underlined number in β and 'Ptr' marked with green or red denotes the start or end index of opinion or role terms, respectively.

role term. β refers to a *buffer* containing unprocessed tokens in X. A is an action *list* to record historical actions. Y stores the generated pairs.

Actions. We design six actions as defined in following:

- **O-START/R-START** means that the top element in β is a start token of an opinion term or a role term. Once determining the start position by the action, the end position will further be obtained by the PointNet, and then the λ representation will be updated.
- **NO-START** implies that the top element in β is not a start token of any term, so pop it out of the β.
- **ARC** builds a valid relation between the top element in σ^* and λ, where $* \in \{o, r\}$. If λ is an opinion term o, we assign a specific role type c for the ⟨o, r⟩ pair, i.e., λ and the top element in σ^r, and pop the top element out of σ^r into α^r. And if λ is a role term r, we decide the role type c for the ⟨o, r⟩ pair, i.e., the top element in σ^o and λ, and pop the top element out of σ^o into α^o.
- **NO-ARC** means there is no valid relation between the top element in $\sigma^*(* \in \{o, r\})$ and λ. If λ is an opinion term,we pop the top element out of σ^r into α^r; Otherwise, we pop the top element out of σ^o into α^o.
- **SHIFT** represents the end of relation detection for λ. So we first pop all elements out of α^* into $\sigma^*(* \in \{o, r\})$. Then, if λ is an opinion term, we pop the element out of λ into σ^o; otherwise we pop the element out of λ into σ^r. Finally, we pop the top element out of the β.

Action Preconditions. We also design several preconditions (Fan et al. 2020) to ensure the valid action candidates to predict at each transition step t. For example, NO/R/O-START can only be conducted when β is not empty and λ is empty.

Linear-time Parsing. Since each unprocessed token in β will be handled only once, i.e., determining whether a token is a start of the opinion or role term, our transition system takes linear-time complexity. Fig. 2 shows an example of how the transition system works. At the initial state, all unprocessed words are stored in β. Then, the states are sequentially changed by a list of actions, during which the opinion terms and their corresponding role terms are extracted into Y. The procedure terminates once both buffer β and λ turn empty. Finally, our system will output all pairs into Y, e.g., total four pairs, '⟨says, He(hd)⟩', '⟨says, the agency(tg)⟩', '⟨seriously needs, the agency(hd)⟩' and '⟨seriously needs, money to develop(tg)⟩', of the sentence in Fig. 2.

4 Neural Transition Model

We construct neural representations for transition states in s, based on which we employ neural networks to predict actions. If an O/R-START action is generated, the PointNet will determine the corresponding end position. And if an ARC action is yielded, the role type detector will assign a role type for the role term in an opinion-role pair.

Word Representation. We consider two types of representation for each word w_i, including a word-level embedding $\boldsymbol{x}_i^w$ from a pre-trained model and a character-level representation $\boldsymbol{x}_i^c$ generated by a convolutional neural network:

$$\boldsymbol{x}_i = [\boldsymbol{x}_i^w; \boldsymbol{x}_i^c] \,. \tag{1}$$

where [;] denotes a concatenation operation.

State Representation. We construct neural state representation $\boldsymbol{e}_t^s$ for the state s_t at step t. We first use a BiLSTM to encode each word in buffer β into representation $\boldsymbol{h}_i$, and also generate the overall β representation $\boldsymbol{e}_t^\beta$. We use another BiLSTM to represent λ and the action list A, i.e., $\boldsymbol{e}_t^\lambda$ and $\boldsymbol{e}_t^A$. We utilize two separate Stack-LSTMs to encode the

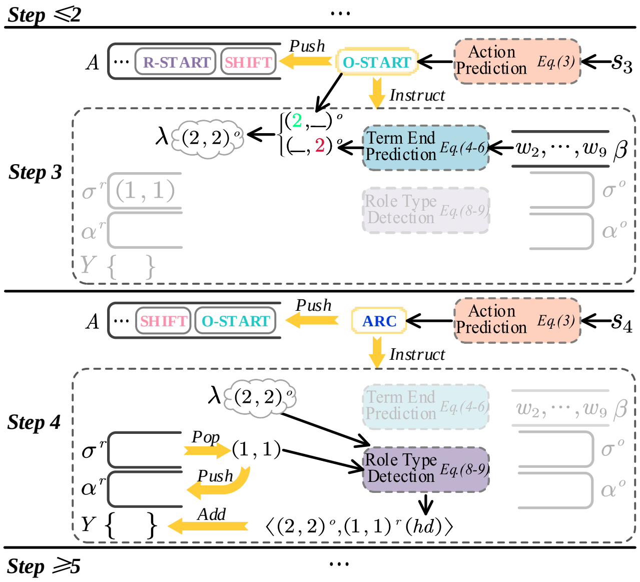

Figure 3: The illustration of transition *step 3* to *step 4* with the vanilla neural model.

stacks σ^o and σ^r, i.e., $\boldsymbol{e}_t^o$ and $\boldsymbol{e}_t^r$. Finally, we summarize all the above items as the overall state representation $\boldsymbol{e}_t^s$:

$$\boldsymbol{e}_t^s = [\boldsymbol{e}_t^\lambda; \boldsymbol{e}_t^o; \boldsymbol{e}_t^r; \boldsymbol{e}_t^A, \boldsymbol{e}_t^\beta] \,, \quad (2)$$

Action Prediction. Based on the state representation $\boldsymbol{e}_t^s$, we first apply a multi-layer perceptron (MLP) on it, and then predict the action y_t^A via a softmax, as shown in Fig. 3:

$$y_t^A = \underset{\mathcal{A}(t)}{\text{Softmax}}(\text{MLP}(\boldsymbol{e}_t^s)) \,, \quad (3)$$

where $\mathcal{A}(t)$ indicates the set of valid action candidates at step t according to the aforementioned preconditions.

Term End Prediction via PointNet. Based on the current start index i of an opinion or role term, as well as the hidden representation $\{\boldsymbol{h}_i, \cdots, \boldsymbol{h}_T\}$ from β, we adopt the PointNet to detect the end index j, as can be found in Fig. 3:

$$u_{ik} = \text{Tanh}(\boldsymbol{W}_1 \boldsymbol{h}_i + \boldsymbol{W}_2 \boldsymbol{h}_k), k = [i, \cdots, T] \,, \quad (4)$$

$$o_{ik} = \text{Softmax}(u_{ik}) \,, \quad (5)$$

$$j = \underset{i \le j \le T}{\text{Argmax}}(o_{ik}) \,, \quad (6)$$

where $\boldsymbol{W}_*$ are trainable parameters (same in below).

After obtaining the start and end index of the term, we generate its representation $\boldsymbol{a}^*(* \in \{o, r\})$ as follows:

$$\boldsymbol{a}^* = \boldsymbol{W}_3[\boldsymbol{h}_i; \boldsymbol{h}_j; \boldsymbol{x}^{P[i:j]}] \quad (7)$$

where $\boldsymbol{x}^{P[i:j]}$ is a term length embedding. The term representation will be employed to update the λ representation.

Role Type Detection via Biaffine. At the meantime, once a valid relation is determined by an ARC action between an opinion term o and a role term r, a biaffine function (Dozat and Manning 2017) assigns a type c for the role (cf. Fig. 3):

$$y^c = \text{Tanh}([\ \boldsymbol{a}^o\ ]^\top \cdot \boldsymbol{W}_4 \cdot \boldsymbol{a}^r) \,, \quad (8)$$

$$c = \text{Softmax}(y^c) \,. \quad (9)$$

By now we can ensemble one complete opinion-role pair $\langle o, r(c) \rangle$, which we will add into Y.

Training Object. The target is to produce the pair structure $\hat{Y}$ given the gold Y, which can be decomposed into three sub-objectives, including action prediction loss $\mathcal{L}_a$, end boundary prediction loss $\mathcal{L}_p$ and role detector loss $\mathcal{L}_c$. We adopt cross-entropy of these predictions over gold labels. We can ensemble the above items:

$$\mathcal{L} = \mathcal{L}_a + \mathcal{L}_p + \mathcal{L}_c + \frac{\eta}{2} \left\| \theta \right\|^2 \,, \quad (10)$$

where θ are the overall model parameters, and η is the coefficient of ℓ_2-norm regularization.

5 Syntax-enhanced Transition Model

The above neural transition model generates a list of actions, from which the final opinion role pairs can be parsed out incrementally. However, the vanilla system takes merely the local-level clues for each action prediction (Li, Zhao, and Parnow 2020), meanwhile without any explicit interaction between opinion and role terms. Therefore, in this section, we consider constructing global-level features and modeling the explicit interaction by integrating the syntax knowledge. Based on the vanilla system, on the one hand we propose a POS-aware PointNet; on the other hand we introduce a dependency-guided high-order interaction mechanism.

5.1 POS-aware PointNet for Term Prediction

Since POS tags can offer potential clues for the term boundary recognition (Nie et al. 2020; Wu et al. 2021), we therefore consider integrating such features into the PointNet. We update the raw pointing calculation in Eq.(4) with

$$u_{ik} = \text{Tanh}(\boldsymbol{W}_5[\boldsymbol{h}_i; \boldsymbol{x}_i^p] + \boldsymbol{W}_6[\boldsymbol{h}_k; \boldsymbol{x}_k^p] + \Delta\text{F}_{k-1,k,k+1}) \,, \quad (11)$$

where we first concatenate a POS embedding $\boldsymbol{x}_i^p$ with token representation $\boldsymbol{h}_i$ for w_i. $\Delta\text{F}_{k-1,k,k+1}$ is a *Boundary Differential Enhancer* (BDE) that captures the boundary differences between the former and the next POS tag of token k:

$$\Delta\text{F}_{k-1,k,k+1} = \boldsymbol{W}_7[(\boldsymbol{x}_k^p - \boldsymbol{x}_{k-1}^p); (\boldsymbol{x}_{k+1}^p - \boldsymbol{x}_k^p)] \,. \quad (12)$$

5.2 Dependency-guided High-order Interaction

Our main motivation is to encourage sufficient interactions between those already-yielded opinion-role structures by the vanilla system so as to better discover the unknown structures. It is possible to directly model the interactions within those opinions and roles that are overlapped together. However those latent interactions between the terms without explicit connections cannot be sufficiently modeled. Thus, we consider leveraging the external syntactic dependency information. Intuitively, the word-word relations depicted by the dependency tree offer much aid for building high-order interactions (Li et al. 2020). Practically, we first construct a unified graph structure to simultaneously model the previously yielded opinion-role structure with the dependency tree globally, i.e., unified dependency-opinion graph (UDOG). We then propose a relation-centered graph aggregator (RCGA) to encode the UDOG, where the resulting high-order representations are further used to promote the predictions in the vanilla transition system.

Building UDOG. Intuitively, the opinion-role structure coincides much with the syntactic dependency structure, and the co-modeling of these two structures would effectively help the interactions between the opinion-role pairs. Given a sentence of words, the corresponding dependency tree, and the currently generated opinion-role structures in Y, the unified dependency-opinion graph can be formalized by G=(V, E). V is a set of word nodes w_i, and E is a set of labeled edges π_v, with E=E_D∪E_C∪E_R. E_D represents the dependency edges attached with syntactic labels, e.g., 'He'$\xleftarrow{\text{nsubj}}$'says'. E_C means the inter-term links, e.g., 'says'$\xrightarrow{\text{tg}}$'the agency'; while E_R means the intra-term links[3], e.g., 'the'$\xleftarrow{\text{role}}$'agency', 'seriously'$\xleftarrow{\text{opn}}$'needs', as shown in Fig. 4(a). Note that UDOG is a multi-relation graph, as a pair of words may be connected with more than one edge. We actively update UDOG once a new opinion-role pair is detected out.

Graph Encoding via RCGA. Existing graph encoders, e.g., graph convolutional network (Marcheggiani and Titov 2017) or graph recurrent network (Zhang and Zhang 2019), have been shown prominently on graph modeling, while they may fail to directly handle our UDOG due to two key characteristics: *labeled edges* and *multi-relational edges*. Thus, we propose a novel relation-centered graph aggregator (RCGA). The key idea is to take the edges as mainstays instead of the neighbor nodes. Technically, we aggregate the information for node w_i from its neighbors as well as the labeled edges:

$$\bar{\boldsymbol{h}}_{i,v} = [\boldsymbol{h}_j; \boldsymbol{x}_{i,v}] \,, \quad j \in \mathcal{B}(i,v) \,, \tag{13}$$

$$\rho_{i,v} = \underset{v\in\mathcal{E}(i)}{\text{Softmax}}(\text{LeakyReLU}(\boldsymbol{W}_8[\boldsymbol{h}_i; \bar{\boldsymbol{h}}_{i,v}])) \,, \tag{14}$$

$$\boldsymbol{h}_i^{'} = \frac{1}{|\mathcal{E}(i)|} \textstyle\sum_{v\in\mathcal{E}(i)} \rho_{i,v} \boldsymbol{W}_9 \bar{\boldsymbol{h}}_{i,v} \,, \tag{15}$$

where $\boldsymbol{x}_{i,v}$ is the label embedding of the edge label π_v, $v \in \mathcal{E}(i)$ is the edge directly connecting with node w_i, and $j \in \mathcal{B}(i,v)$ is the target node of edge v. $\bar{\boldsymbol{h}}_{i,v}$ is the relation-centered representation that entails both neighbor node and the edge label information. $\rho_{i,v}$ is the neighboring matrix. In particular we take 2 iterations of RCGA propagation for a thorough structure communication. Also, RCGA performs above calculations whenever the UDOG is updated. Fig. 4(b) illustrates the RCGA based on w_6 as the current node.

Updating Predictions. Once obtaining the enhanced node representations $\{\boldsymbol{h}_1^{'}, \cdots, \boldsymbol{h}_T^{'}\}$, we then update the predictions in the vanilla systems. We first enhance the action prediction by additionally adding a global-level feature representation $\bar{\boldsymbol{g}}$ into the original $\boldsymbol{e}_t^s$:

$$\begin{aligned} \bar{\boldsymbol{g}} &= \text{Graph-Pooling}(\{\boldsymbol{h}_1^{'}, \cdots, \boldsymbol{h}_T^{'}\}) \,, \\ \boldsymbol{e}_t^s &:= [\boldsymbol{e}_t^s; \bar{\boldsymbol{g}}] \,, \end{aligned} \tag{16}$$

where Graph-Pooling is the max-pooling operation over the RCGA. At the meantime, we also replace the original token representation $\boldsymbol{h}_i$ used in Eq.(11) with the updated one $\boldsymbol{h}_i^{'}$.

Finally, we reinforce the first-order biaffine role type detector in Eq.(8) with a high-order triaffine attention:

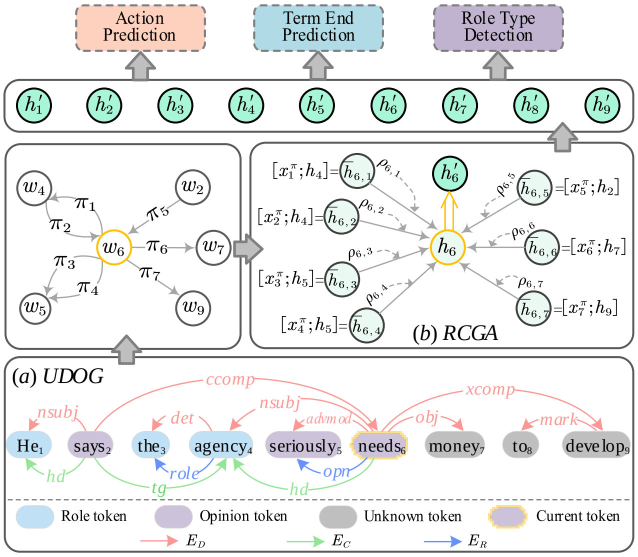

Figure 4: Encoding UDOG (a) via RCGA (b). The resulting high-order representation is used for action prediction, term end prediction, and role type detection.

$$y^c = \text{Sigmoid}(\begin{bmatrix} \bar{\boldsymbol{a}}^o \\ 1 \end{bmatrix}^{\mathsf{T}} (\bar{\boldsymbol{a}}^r)^{\mathsf{T}} \mathbf{W}_{10} \begin{bmatrix} \bar{\boldsymbol{g}} \\ 1 \end{bmatrix}) \,, \tag{17}$$

where $\bar{\boldsymbol{a}}^*$ is the updated term representations. Intuitively, the role type prediction can be improved in the help of such global-level feature.

6 Experiment

6.1 Setup

Datasets. We experiment with the ORL benchmark, MPQA v2.0. The dataset has a fixed number of 132 documents for developing. And for the rest of 350 documents, following the prior work (Zhang et al. 2020; Xia et al. 2021), we take five-fold cross-validation, i.e., splitting 350 documents into 280 and 70 documents as training and testing, respectively.

Implementation. We use the pre-trained 300-d GloVe (Pennington et al. 2014) for word-level embedding, and BERT (base cased version) also is additionally used (Devlin et al. 2019). The kernel sizes of CNN for character-level representations are [3,4,5]. The hidden size of BiLSTM is 150, and the Stack-LSTM is 300. Other embedding sizes are all set to 50. We denote the vanilla neural transition model as 'PtrTrans', and the syntax-enhanced version as 'SyPtrTrans'.

Baselines. We make comparisons with the existing end-to-end ORL baselines. 1) **BiLSTM+CRF**: Katiyar et al. (2016) propose a joint ORL model based on an enhanced sequence labeling scheme. 2) **Transition** model for joint ORL (Zhang et al. 2019). 3) **SPANOM**: a span-based graph model (Xia et al. 2021). Also, **SPANOM+Syn** is the version that integrates syntactic constituent features.

Evaluation. We follow Xia et al.(2021), taking the **exact F1**, **binary F1** and **proportional F1** scores as the evaluating metrics. We measure the following prediction objects (or

[3] We construct the intra-term arcs by a *tail-first* strategy (Barnes et al. 2021), connecting tail token to all the other tokens.

	$\mathcal{O}$	$\mathcal{O}$-$\mathcal{R}$	$\mathcal{O}$-$\mathcal{R}(hd)$	$\mathcal{O}$-$\mathcal{R}(tg)$
• w/o BERT				
BiLSTM+CRF[†]	52.65	-	46.72	30.07
Transition[†]	53.04	-	47.02	31.45
SPANOM[†]	58.06	43.12	52.90	32.42
PtrTrans	58.14	43.66	53.19	33.23
SyPtrTrans	**59.87**	**44.41**	**54.67**	**35.03**
• w/ BERT				
SPANOM[†]	63.71	49.89	58.24	41.10
SPANOM+Syn[†]	-	50.46	58.46	41.82
PtrTrans	63.90	50.11	58.28	41.96
SyPtrTrans	**65.28**	**51.62**	**59.48**	**44.04**

Table 1: Main results in exact F1 scores. Baselines with the superscript '†' are copied from Xia et al. (2021).

	Binary F1		Proportional F1	
	$\mathcal{O}$-$\mathcal{R}(hd)$	$\mathcal{O}$-$\mathcal{R}(tg)$	$\mathcal{O}$-$\mathcal{R}(hd)$	$\mathcal{O}$-$\mathcal{R}(tg)$
• w/o BERT				
BiLSTM+CRF[†]	58.22	54.98	-	-
Transition[†]	**60.93**	**56.44**	-	-
SPANOM[†]	56.47	45.09	55.62	41.65
PtrTrans	57.91	46.96	56.88	42.82
SyPtrTrans	58.29	47.68	**57.12**	**43.63**
• w/ BERT				
BERT+CRF[†]	55.52	50.39	46.62	34.29
SPANOM[†]	62.04	53.27	61.20	49.88
PtrTrans	62.25	53.66	61.65	50.42
SyPtrTrans	**63.22**	**55.23**	**62.34**	**52.02**

Table 2: The results in binary and proportional F1 scores.

	$\mathcal{O}$-$\mathcal{R}(hd)$ / Δ	$\mathcal{O}$-$\mathcal{R}(tg)$ / Δ
SyPtrTrans	**59.48**	**44.04**
• Input features		
w/o Char ($\boldsymbol{x}_i^c$)	59.18 / -0.30	43.75 / -0.29
• Boundary detection features		
w/o BDE (Eq.12)	59.32 / -0.16	43.89 / -0.15
w/o POS	59.20 / -0.28	43.48 / -0.56
• High-order Interaction features		
w/o Opn	58.85 / -0.63	42.99 / -1.05
w/o Dep	58.72 / -0.76	42.58 / -1.46
w/o UDOG	58.56 / -0.92	42.13 / -1.91

Table 3: The ablation results (exact F1). 'w/o Opn' means building UDOG without using the opinion-role structure, and 'w/o Dep' without using dependency structure.

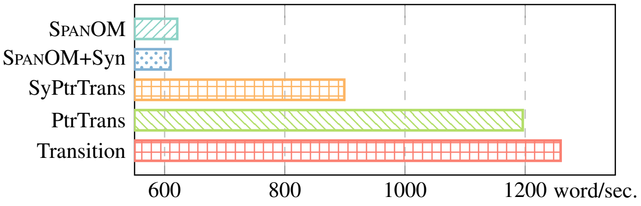

Figure 5: Comparisons on decoding speed.

subtasks): 1) the opinion terms ($\mathcal{O}$), where a term is correct when both the start and end boundaries of a term are correct; 2) the opinion-role pairs, including the roles of *holder* type ($\mathcal{O}$-$\mathcal{R}(hd)$), the *target* type ($\mathcal{O}$-$\mathcal{R}(tg)$), and the overall both two type ($\mathcal{O}$-$\mathcal{R}$). A pair prediction is correct only when the opinion term, role term and role type are all correct. All results of our two models are presented after a significant test with the best baselines, with $p \leq 0.05$.

6.2 Results and Discussion

Main Result. Table 1 reports the results of different models. Firstly let's check the performances with GloVe embedding. We see that PtrTrans is superior to all baselines, especially outperforming the best-performing baseline, SPANOM. Even based on the homologous transition technique, our PtrTrans significantly beats the one of Zhang et al. (2019). This is largely because our transition system can deal with the term overlap issues; and meanwhile our PointNet design empowers more accurate term extraction. More crucially, our syntax-enhanced SyPtrTrans model brings the most prominent results against all baselines on total of four subtasks.

Further, with the help of BERT, all systems obtain universally boosted performances. Also with the integration of external syntax information, the results can be improved. For example, even the syntax-aware baseline SPANOM+Syn surpasses our PtrTrans model on $\mathcal{O}$-$\mathcal{R}$ pair detection. Nevertheless, our SyPtrTrans system still keeps its great advantage, e.g., giving the best exact F1 scores, i.e., 65.28% on opinion term extraction and 51.62% on the $\mathcal{O}$-$\mathcal{R}$.

In Table 2 we additionally present the performances under binary F1 and proportional F1 metrics. We observe that the weak baselines (e.g., Transition, BiLSTM+CRF) with GloVe embedding achieve higher binary F1 scores. The underlying reasons can be that these models tend to give more extractions on the shorter text fragments, and thus result in higher scores under the 'binary measurement'. In contrast, our systems, due to the equipment of PointNet, can successfully manage the detection of terms, especially including the longer ones. Detailed analysis is presented later.

Ablation Study. We can first learn from Table 1 and Table 2 that the BERT contextualized word embedding shows the greatest impacts to the system, demonstrating its extraordinary usefulness to the semantics capturing. Table 3 shows the ablation of our best SyPtrTrans model in terms of other factors. We can see that the character feature as well as the *Boundary Differential Enhancer* mechanism show some certain impacts to the system, while stopping injecting the POS tag features give worse performances for the term boundary detection. Looking into the syntax-aided high-order interaction, we observe that the unavailability of either the opinion-role structure or the syntactic dependency structure could lead to sub-optimal task prediction, while the whole removal of UDOG substantially deteriorates the overall performances due to no modeling the interactions among the opinion and role terms.

Decoding Efficiency. In Fig. 5 we compare the decoding speed of different systems. We observe that these transition-

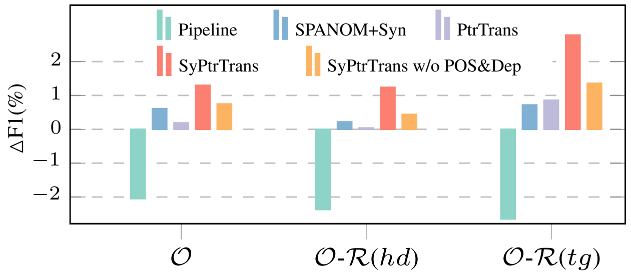

Figure 6: Performance drops of an ORL system comparing to SPANOM. We re-implement a BERT+sequence-labeling model as the *Pipeline* baseline: i.e., first extracting terms and then role type relation classification.

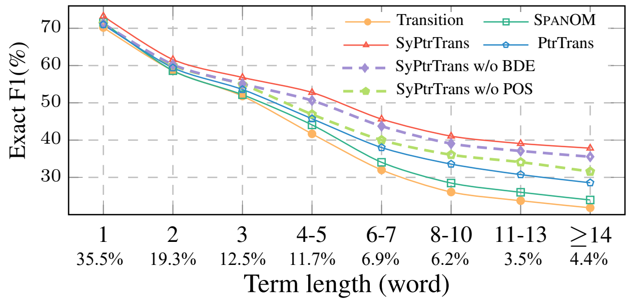

Figure 7: Results of term extraction by varying term lengths.

based methods show very significant model efficiency, due to their linear-time complexity characteristics. Our PtrTrans achieves 2× faster decoding than the span-based graph models (i.e., SPANOM or with syntax enhancement), due to their iterative enumerating nature. We can also notice that the Transition baseline (Zhang et al. 2019) runs slightly faster than ours, partially due to the more sophisticated design of our model. Besides, integrating the external syntax information also could considerably degrade the inference speed, i.e., PtrTrans vs. SyPtrTrans.

Study of the Interaction. We investigate the impact of interaction mechanisms for ORL, as shown in Fig. 6. First of all, the pipeline system without any interaction between two stages of subtasks results in a severe performance drop compared to the joint method of SPANOM. Given that the unified ORL modeling naturally entails certain implicit interactions, our PtrTrans still achieves slightly higher results than SPANOM or even with syntax aided. Notedly, we see that with the explicit interaction from co-modeling of opinion-role structure and syntax structure, our SyPtrTrans can significantly outperform those systems with 'implicit interaction'. Even the explicit interaction is merely from the partial opinion structure, i.e., without integrating external knowledge (POS&dependency), SyPtrTrans still gives higher performances than others.

Impact of Term Length. We further study the impact of different role term lengths. From the trends in Fig. 7 we see that overall, the performances decrease with the increased term length. Also, due to span term enumerating characteristics, the span-based graph model SPANOM performs slightly better than the Transition baseline. In the contrast, our transition frameworks with the design of pointer network are more extraordinary than any baseline on handling terms at any length, especially on the longer terms. More prominently, the additional boundary features from POS contribute a lot to fighting against the challenge of long-length terms, among which the proposed *boundary differential enhancing* mechanism also benefits to some certain extent.

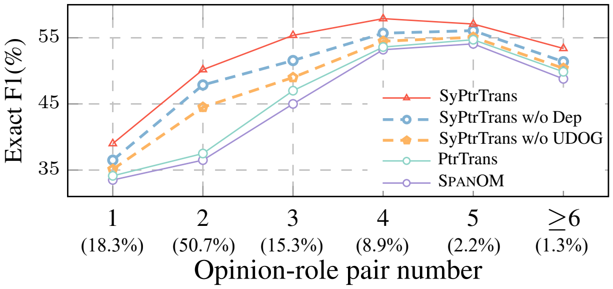

Figure 8: Performances ($\mathcal{O}$-$\mathcal{R}$) with different opinion-role pair number in a sentence.

Influences of the Opinion-role Pair Numbers. We finally in Fig. 8 plot the results with different numbers of opinion-role pairs in a sentence. We observe the essential pattern of ORL task that, fewer co-existence of opinion and role pairs (i.e., ≤2) can actually lead to lower recognizing performances of all models, comparing to the results with more opinion-role pairs in sentences. The main cause is that too few opinion-role pairs result in positive label sparsity, while more pair co-occurrence (i.e., ≥4) provides rich training signals and enables better learning. Under such a circumstance, the model considering rich interactions of opinion-role pairs (e.g., our SyPtrTrans via UDOG) could consequently enhance the task performances, comparing to those without explicit interactions (e.g., PtrTrans and SPANOM). Also from the trends, we see that the co-modeling of the syntactic dependency structure for building the UDOG is quite important to the high-order interaction.

7 Conclusion

In this work we present a novel unified opinion role labeling framework by implementing a neural transition architecture with a pointer network (PointNet). The system enables more accurate predictions on longer terms, and solving the overlap issue meanwhile advancing in a linear-time decoding efficiency. We then enhance the PointNet for boundary detection by integrating linguistic POS tag features. We further explore an explicit interaction between the opinion and role terms by co-modeling the syntactic dependency structure and the partial opinion-role structure as a unified graph, which is further encoded via a novel relation-centered graph aggregator. Experimental results demonstrate that our vanilla transition system outperforms the top-performing baseline, meanwhile achieving 2× faster decoding. By capturing rich high-order features via explicit interaction, our syntax-enhanced system obtains new state-of-the-art results on the benchmark dataset.

Acknowledgments

This work is supported by the National Natural Science Foundation of China (No.61772378, No. 62176187), the National Key Research and Development Program of China (No. 2017YFC1200500), the Research Foundation of Ministry of Education of China (No.18JZD015).

References

Barnes, J.; Kurtz, R.; Oepen, S.; Øvrelid, L.; and Velldal, E. 2021. Structured Sentiment Analysis as Dependency Graph Parsing. In *Proceedings of the ACL-IJCNLP*, 3387–3402.

Breck, E.; Choi, Y.; and Cardie, C. 2007. Identifying Expressions of Opinion in Context. In *Proceedings of the IJCAI*, 2683–2688.

Chen, H.-Y.; and Chen, H.-H. 2016. Implicit Polarity and Implicit Aspect Recognition in Opinion Mining. In *Proceedings of the ACL*, 20–25.

Devlin, J.; Chang, M.-W.; Lee, K.; and Toutanova, K. 2019. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. In *Proceedings of the NAACL*, 4171–4186.

Dozat, T.; and Manning, C. D. 2017. Deep Biaffine Attention for Neural Dependency Parsing. In *Proceedings of the ICLR*.

Dyer, C.; Ballesteros, M.; Ling, W.; Matthews, A.; and Smith, N. A. 2015. Transition-Based Dependency Parsing with Stack Long Short-Term Memory. In *Proceedings of the ACL*, 334–343.

Fan, C.; Yuan, C.; Du, J.; Gui, L.; Yang, M.; and Xu, R. 2020. Transition-based Directed Graph Construction for Emotion-Cause Pair Extraction. In *Proceedings of the ACL*, 3707–3717.

Fei, H.; Ren, Y.; and Ji, D. 2020. Boundaries and edges rethinking: An end-to-end neural model for overlapping entity relation extraction. *Information Processing & Management*, 57(6): 102311.

Fei, H.; Zhang, M.; Li, B.; and Ji, D. 2021. End-to-end Semantic Role Labeling with Neural Transition-based Model. In *Proceedings of the AAAI*, 12803–12811.

Fernández-González, D.; and Gómez-Rodríguez, C. 2020. Transition-based Semantic Dependency Parsing with Pointer Networks. In *Proceedings of the ACL*, 7035–7046.

Fu, J.; Huang, X.; and Liu, P. 2021. SpanNER: Named Entity Re-/Recognition as Span Prediction. In *Proceedings of the ACL*, 7183–7195.

Hu, M.; and Liu, B. 2004. Mining Opinion Features in Customer Reviews. In *Proceedings of the AAAI*, 755–760.

Kannangara, S. 2018. Mining Twitter for Fine-Grained Political Opinion Polarity Classification, Ideology Detection and Sarcasm Detection. In Chang, Y.; Zhai, C.; Liu, Y.; and Maarek, Y., eds., *Proceedings of the WSDM*, 751–752.

Katiyar, A.; and Cardie, C. 2016. Investigating LSTMs for Joint Extraction of Opinion Entities and Relations. In *Proceedings of the ACL*, 919–929.

Kim, S.-M.; and Hovy, E. 2006. Extracting Opinions, Opinion Holders, and Topics Expressed in Online News Media Text. In *Proceedings of the Workshop on Sentiment and Subjectivity in Text*, 1–8.

Li, F.; Lin, Z.; Zhang, M.; and Ji, D. 2021. A Span-Based Model for Joint Overlapped and Discontinuous Named Entity Recognition. In *Proceedings of the ACL*, 4814–4828.

Li, Z.; Zhao, H.; and Parnow, K. 2020. Global Greedy Dependency Parsing. In *Proceedings of the AAAI*, 8319–8326.

Li, Z.; Zhao, H.; Wang, R.; and Parnow, K. 2020. High-order Semantic Role Labeling. In *Findings of the EMNLP(findings)*, 1134–1151.

Lin, Y.; Yang, S.; Stoyanov, V.; and Ji, H. 2018. A Multi-lingual Multi-task Architecture for Low-resource Sequence Labeling. In *Proceedings of the ACL*, 799–809.

Marasović, A.; and Frank, A. 2018. SRL4ORL: Improving Opinion Role Labeling Using Multi-Task Learning with Semantic Role Labeling. In *Proceedings of the NAACL*, 583–594.

Marcheggiani, D.; and Titov, I. 2017. Encoding Sentences with Graph Convolutional Networks for Semantic Role Labeling. In *Proceedings of the EMNLP*, 1506–1515.

Nie, Y.; Tian, Y.; Song, Y.; Ao, X.; and Wan, X. 2020. Improving Named Entity Recognition with Attentive Ensemble of Syntactic Information. In *Proceedings of the EMNLP*, 4231–4245.

Pang, B.; and Lee, L. 2007. Opinion Mining and Sentiment Analysis. *Foundations and Trends in Information Retrieval*, 2(1-2): 1–135.

Pennington, J.; Socher, R.; and Manning, C. 2014. GloVe: Global Vectors for Word Representation. In *Proceedings of the EMNLP*, 1532–1543.

Tang, F.; Fu, L.; Yao, B.; and Xu, W. 2019. Aspect based fine-grained sentiment analysis for online reviews. *Information Science*, 488: 190–204.

Vinyals, O.; Fortunato, M.; and Jaitly, N. 2015. Pointer Networks. In *Proceedings of the NeurIPS*, 2692–2700.

Wiebe, J.; Wilson, T.; and Cardie, C. 2005. Annotating Expressions of Opinions and Emotions in Language. *Language Resources and Evaluation*, 39(2-3): 165–210.

Wu, S.; Fei, H.; Ren, Y.; Ji, D.; and Li, J. 2021. Learn from Syntax: Improving Pair-wise Aspect and Opinion Terms Extraction with Rich Syntactic Knowledge. In *Proceedings of the IJCAI*, 3957–3963.

Xia, Q.; Zhang, B.; Wang, R.; Li, Z.; Zhang, Y.; Huang, F.; Si, L.; and Zhang, M. 2021. A Unified Span-Based Approach for Opinion Mining with Syntactic Constituents. In *Proceedings of the NAACL*, 1795–1804.

Yang, B.; and Cardie, C. 2012. Extracting Opinion Expressions with semi-Markov Conditional Random Fields. In *Proceedings of the EMNLP*, 1335–1345.

Zeng, X.; Zeng, D.; He, S.; Liu, K.; and Zhao, J. 2018. Extracting Relational Facts by an End-to-End Neural Model with Copy Mechanism. In *Proceedings of the ACL*, 506–514.

Zhang, B.; Zhang, Y.; Wang, R.; Li, Z.; and Zhang, M. 2020. Syntax-Aware Opinion Role Labeling with Dependency Graph Convolutional Networks. In *Proceedings of the ACL*, 3249–3258.

Zhang, M.; Liang, P.; and Fu, G. 2019. Enhancing Opinion Role Labeling with Semantic-Aware Word Representations from Semantic Role Labeling. In *Proceedings of the NAACL*, 641–646.

Zhang, M.; Wang, Q.; and Fu, G. 2019. End-to-end neural opinion extraction with a transition-based model. *Information System*, 80: 56–63.

Zhang, Y.; and Clark, S. 2010. A Fast Decoder for Joint Word Segmentation and POS-Tagging Using a Single Discriminative Model. In *Proceedings of the EMNLP*, 843–852.

Zhang, Y.; and Zhang, Y. 2019. Tree Communication Models for Sentiment Analysis. In *Proceedings of the ACL*, 3518–3527.