Proceedings of the Thirtieth International Joint Conference on Artificial Intelligence (IJCAI-21)
Sister Conferences Best Papers Track

Open Intent Extraction from Natural Language Interactions (Extended Abstract)*

Nikhita Vedula[1†], **Nedim Lipka**[2], **Pranav Maneriker**[3] and **Srinivasan Parthasarathy**[3]
[1]Amazon
[2]Adobe Research
[3]Ohio State University
{vedula.5, maneriker.1, parthasarathy.2}@osu.edu, lipka@adobe.com

Abstract

Accurately discovering user intents from their written or spoken language plays a critical role in natural language understanding and automated dialog response. Most existing research models this as a classification task with a single intent label per utterance. Going beyond this formulation, we define and investigate a new problem of *open intent* discovery. It involves discovering one or more generic intent types from text utterances, that may not have been encountered during training. We propose a novel, domain-agnostic approach, *OPINE*, which formulates the problem as a sequence tagging task in an open-world setting. It employs a CRF on top of a bidirectional LSTM to extract intents in a consistent format, subject to constraints among intent tag labels. We apply multi-headed self-attention and adversarial training to effectively learn dependencies between distant words, and robustly adapt our model across varying domains. We also curate and release an intent-annotated dataset of 25K real-life utterances spanning diverse domains. Extensive experiments show that OPINE outperforms state-of-art baselines by 5-15% F1 score.

1 Introduction and Background

Recent advances in natural language understanding (NLU) and speech recognition have triggered the advent of a wealth of conversational agents such as Apple's Siri and Amazon's Alexa. Such agents need to parse and interpret human utterances, especially people's intentions or *intents*, and respond accordingly. Most existing work [Chen *et al.*, 2013; Gupta *et al.*, 2014; Wang *et al.*, 2015; Kim *et al.*, 2016; Liu and Lane, 2016; Zhang and Wang, 2016; Kim *et al.*, 2017; Coucke *et al.*, 2018; Xia *et al.*, 2018] detects user intents via multi-class classification, by categorizing input utterances into pre-defined intent classes for which sufficient labeled data is available during model training. We define a novel task of identifying and extracting explicit user intents from text utterances in an *open-world* setting, without any prior knowledge of the intent classes that the text may contain, and name it *Open Intent Discovery*. We propose a framework called OPINE (OPen INtent Extraction) [Vedula *et al.*, 2020] to solve this task. It can recognize instances of novel or newly emerging intent types at test time that it has never seen before during model training. [Xia *et al.*, 2018] solve a similar problem using zero-shot classification but assume that the list of new or unseen (during training) intent classes is available at test time along with some knowledge about them. Yet other techniques [Kim and Kim, 2018; Lin and Xu, 2019] can only identify if an input utterance is likely to contain a new intent or domain. They do not 'discover' or specify what the new intents are. Further, the above mentioned approaches cannot detect more than one intent within an input utterance. To the best of our knowledge, our work is the *first* to address the above limitations.

Unlike prior work, OPINE models open intent discovery as a *sequence tagging* task (Section 2). We develop a neural model consisting of a Conditional Random Field (CRF) on top of a bidirectional LSTM with a multi-head self-attention mechanism. OPINE represents all types of user intents in a consistent, generalizable and domain-agnostic format. We also employ adversarial training at the lower layers of our model, and unsupervised pre-training in the target domain under consideration. Commonly used datasets in the intent detection literature such as SNIPS [Coucke *et al.*, 2018] or ATIS [Dahl *et al.*, 1994] largely have concise, coherent and single-sentence texts. They are not very representative of complex, real-world dialog scenarios which could be verbose and ungrammatical, with intents scattered throughout their content. Thus, we also publish a large dataset with 25K real-world utterances, human-annotated with intents, from the online Stack Exchange[1] forum.

2 Problem Formulation

The objective of the *Open Intent Discovery* task is to identify all possible *actionable intents* from text utterances. These may be underlying goals, activities or tasks that a user wants to perform or have performed. We define an *intent* as consisting of two parts [Wang *et al.*, 2015; Chen *et al.*, 2013]:

*This is an abridged version of the paper [Vedula *et al.*, 2020] that won the Best Paper Award at *The Web Conference*, 2020.

†Work done at Adobe Research and the Ohio State University.

[1]www.stackexchange.com

(i) an *action*, which is a word or phrase representing a tangible purpose, task or activity which is to be requested or performed, and (ii) an *object*, which represents those entity words or phrases that the action is going to act or operate upon. We choose such a definition for user intents to address commonly available user interactions within help or customer support forums and with smart speaker devices. For instance, the intent of the text *"Please make a 10:30 sharp appointment for a haircut"* is to make or schedule a haircut appointment. It consists of an action *"make"* and an object *"appointment"*, *"appointment for haircut"*, or *"haircut appointment"*. User utterances that indicate an intent by implying an object, without explicitly mentioning it are outside the scope of this work. We then formulate the open intent discovery problem as a sequence tagging task over three tags: ACTION, OBJECT, and NONE (the remaining words that are neither an ACTION nor an OBJECT). A user intent consists of a matching pair of an ACTION phrase and an OBJECT phrase.

The Open Intent Discovery task differs from the Open Information Extraction (OpenIE) (e.g. [Angeli *et al.*, 2015]) and Semantic Role Labeling (SRL) tasks (e.g. [Tan *et al.*, 2018]) as follows: (i) OpenIE is used to extract relation triples, with the constituents occurring in the input sentence, whereas we define intents as ACTION-OBJECT pairs. (ii) SRL labels and relate constituents in input sentences with their semantic meanings. Not all such constituents pertain to expressed user intent; we focus on intent relations only. (iii) Typical OpenIE and SRL solutions use individual sentences as inputs. OPINE does not have this restriction, and can distinguish sentences with extraneous information that do not express users' intent.

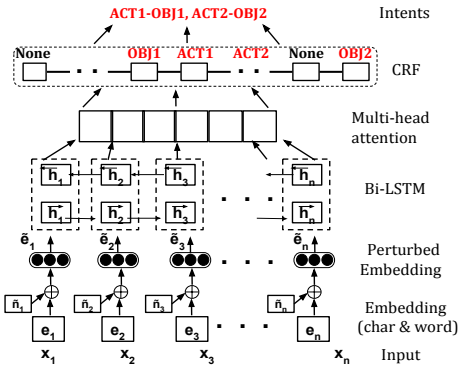

Figure 1: Our OPINE open intent extraction model

3 The OPINE Framework

Figure 1 displays the architecture of OPINE. Given an input text x consisting of a sequence of words $[x_1, x_2, ..., x_n]$, we first transform it into a feature sequence by constructing the character level representation of each word x_i, using a CNN [Huang *et al.*, 2015]. We use a *highway network* [Srivastava *et al.*, 2015] to combine the character level embeddings with pre-trained word level embeddings in a balanced manner, to obtain embedding e_i for every word x_i. This is input to the next layer, namely a bidirectional LSTM [Hochreiter and Schmidhuber, 1997; Graves *et al.*, 2013] that generates a sequence of representations $[h_1, h_2, ..., h_n]$ from forward and backward sequence contexts.

Adversarial Training. We generate *adversarial* input examples that are very close to the original inputs and should yield the same labels, by adding small, continuous, worst case perturbations or noise to the embedding layer in the direction that significantly increases the model's loss function. We then train OPINE on the mix of original and adversarial examples to regularize our model [Goodfellow *et al.*, 2015; Miyato *et al.*, 2016], improve its robustness to slight input perturbations, and discover features and structures common across multiple domains. Let input text $x = [x_1, \ldots, x_n]$ be represented by embedding e. We generate its worst case perturbation η of a small bounded norm ϵ, which is a tunable hyperparameter. We use the first order approximation via the fast gradient method [Goodfellow *et al.*, 2015] to obtain an approximate worst case perturbation of norm ϵ. We also normalize the word and character embeddings, so that the model does not trivially learn the embeddings of large norms and make the perturbations insignificant [Miyato *et al.*, 2016].

$$\widetilde{\eta} = \epsilon \frac{g}{||g||}; \text{ where } g = \nabla_e(\mathcal{L}(e;\theta))$$
$$\widetilde{e} = e + \widetilde{\eta}$$
$$\mathcal{L}' = \alpha\mathcal{L}(e;\theta') + (1-\alpha)\mathcal{L}(\widetilde{e};\theta')$$

Here $\widetilde{e}$ represents the perturbed embedding of an adversarial example generated from embedding e and ∇_e denotes the gradient operator. $\mathcal{L}(e;\theta')$ and $\mathcal{L}(\widetilde{e};\theta')$ represent the loss functions from the original training instance and its adversarial transformation respectively. α is a weighting parameter. The new loss function $\mathcal{L}'$ can be optimized in the same way as the original loss $\mathcal{L}$. While generating adversarial examples, we measure the semantic (cosine) similarity between the original and adversarial embeddings, and only choose those examples where the similarity is greater than a threshold.

Attention Mechanism. We employ a multi-head self-attention mechanism [Vaswani *et al.*, 2017] to select and focus on the important and essential hidden states of the Bi-LSTM layer. It jointly attends to information at different positions of the input sequence with multiple individual attention functions and separately normalized parameters called *attention heads*. This enables it to capture different contexts in a fine-grained manner and learn long-range dependencies effectively. Each attention head computes a sequence z from the output $h = [h_1, h_2, ..., h_n]$ of the Bi-LSTM layer.

Sequence Tagging via Constraint-enhanced CRFs. The output of the attention layer $z = [z_1, z_2, ..., z_n]$ serves as input to the next layer of OPINE, namely a linear chain CRF with maximum conditional log-likelihood estimation [Lafferty *et al.*, 2001]. It predicts one of three tags for each word of the input sequence: ACTION, OBJECT, or NONE. We impose the following additional constraints on the Viterbi decoding algorithm [Forney, 1973] of the CRF. First, as per our definition of a valid intent, we want to ensure that the CRF never predicts only an ACTION tag or only an OBJECT tag. Next, we identify a small number of *intent indicator* phrases that suggest the presence of an intent in the text surrounding them [Gupta *et al.*, 2014; Wang *et al.*, 2015]. For each

such phrase, we selectively choose candidates having labelled intent tags in a small contextual neighbourhood (up to five words) following the intent indicator. We apply these constraints in two ways during the CRF tag inference phrase: The first is using a *beam search* that penalizes sequences in the beam not satisfying the constraints and falls back to using the next most probable tag predictions. Second, we reduce the solution output by the Viterbi algorithm to a shortest path problem in a graph constructed among the sequence tokens and the possible tag values each token can take [Roth and Yih, 2005]. We solve this using Integer Linear Programming with added tag-specific constraints to it as inequalities between the graph node variables.

Generating Intents from Tag Sequences. Once the CRF predicts ACTION, OBJECT and NONE tags for each input word, our final step is to combine ACTION tagged phrases followed by OBJECT tagged phrases to generate meaningful intents. We develop two techniques for this. First, we employ the simple but effective distance-based heuristic of linking ACTION and OBJECT tagged phrases with respect to their word-based proximity in the input text. Our second technique learns a multi-layer perceptron (MLP) classifier. The input features for the MLP consist of the sum of the pre-trained GloVe embeddings [Pennington *et al.*, 2014] of the words in the potential ACTION-OBJECT intent phrase, concatenated with the normalized word distance value between the ACTION and OBJECT phrases in the original input text. The MLP contains two fully connected ReLU layers, followed by a fully connected layer of size one. It outputs the probability of combining each potential ACTION-OBJECT pair under consideration, to produce an intent. The MLP is trained with a margin-based hinge loss function, maximizing the separation between the true and the highest scoring incorrect OBJECT option for the current ACTION phrase.

4 Evaluation

Data Collection. We collected about 75K questions with their top correct answer on various topical categories, from www.stackexchange.com, due to its long and verbose text with background details, linguistic complexity and diversity, and multiple intents scattered throughout the text. We formulated an Amazon Mechanical Turk experiment to annotate 25K of these with up to three intents that the crowd workers felt were most important or relevant (inter-annotator agreement = 0.73). Our intent-annotated dataset consists of 12 diverse genres (e.g. *DIY, Life Hacks, Data Science, WebApps*) and hundreds of unique intents. We have made this dataset publicly available for future research on this topic. We used the remaining 50K unlabeled questions for unsupervised pre-training, by generating *verb-object* parse tags for them via the Stanford CoreNLP dependency parser [Manning *et al.*, 2014], and using these words as proxies for the ACTION and OBJECT tagged phrases that compose an intent.

4.1 Results

Table 1 shows the performance of various baseline approaches for open intent extraction on our curated Stack Exchange dataset. The first baseline levers a cue-based intent detection strategy [Gupta *et al.*, 2014; Wang *et al.*, 2015] that essentially returns as intents the phrases following the occurrence of 'intent-indicator' cue words or phrases (described in Section 3). The second baseline levers the verb-object tags learned by the Stanford dependency parser, used as proxies for ACTION and OBJECT tags respectively. The third approach is a state-of-the-art deep semantic role labeling model with self attention [Tan *et al.*, 2018], for which we only use the two roles of verb and the object or entity acted upon by the verb as contributors to user intent. . The last column of semantic similarity computes the mean of the cosine similarities between the embeddings of the predicted and actual (annotated) intents. Each intent phrase's embedding is the average of the pre-trained GloVe [Pennington *et al.*, 2014] embeddings of its constituent words. '*beam-CRF*' and '*constr-CRF*' in the last two rows refer to (i) considering a beam of probable tag sequences, and (ii) adding additional constraints to the decoding algorithm, from Section 3. '*att*' and '*adv*' denote the presence of attention and adversarial training respectively. *'w-dist'* denotes the word-distance heuristic of matching ACTION-OBJECT phrases to create an intent.

Pre-training our model with the dependency parser data followed by fine-tuning on the intent-labeled data improves the F1-score by at least 6%. Enhancing the CRF decoding algorithm with constraints (*beam-CRF* and *constr-CRF*) benefits the F1-score further by 2-5%. OPINE significantly outperforms the simple intent-indicator based model and the Stanford parser (first two rows of Table 1) by over 15%, and the SRL model (third row of Table 1) by about 9% in terms of F1-score and semantic similarity. This shows that OPINE can successfully filter out all the "non-intent" background information present in the input utterance, and only focus on the user intent text. Overall, OPINE as proposed outperforms all baselines, with an intent F1 score of 76%, and a semantic similarity of 86% between the true and predicted intents.

Domain Adaptation Capability. We investigate OPINE's capability in adapting and transferring knowledge across distinct conversational domains d. We exclude all utterances from each domain d while training OPINE, and directly test on utterances from d. The average difference in F1-score and semantic similarity metrics with and without using training data from the test domain d is $\leq 5\%$. Only the *Life Hacks* domain suffers a loss of 6.5% in terms of F1-score. Interestingly for the *DIY* domain, its training data is dominated by other semantically distinct domains. However, OPINE still attains a good F1-score of 72%, only 4% lesser than what is possible if *DIY* domain data is used for training. These results show that OPINE can effectively detect novel actionable intents in low-resource domains with minimal manual effort.

Role of Attention. We find that the presence of attention lends OPINE an F1 score gain of at least 4%. We examine and visualize in Table 2 the self-attention values for truncated versions of specific utterances and their intents from our Stack Exchange dataset. A darker colored highlight on a word indicates that it receives higher attention, and plays a greater role in intent discovery. Words constituting intents are highlighted in boldface. In all cases, we observe that words semantically related to and contributing to at least one intent are success-

Approach	ACTION P/R/F1	OBJECT P/R/F1	Intent P/R/F1	Sim.
Cue-based Intents [Gupta *et al.*, 2014] [Wang *et al.*, 2015]	0.65/0.59/0.62	0.6/0.54/0.57	0.63/0.56/0.59	0.67
Stanford CoreNLP (SC) parser [Manning *et al.*, 2014]	0.56/0.49/0.52	0.51/0.43/0.47	0.53/0.45/0.49	0.59
Deep Semantic Role Labeling (SRL) [Tan *et al.*, 2018]	0.79/0.63/0.7	0.69/0.62/0.65	0.7/0.62/0.66	0.75
att + SC (pre-train) + MTurk (fine tune) + w-dist	0.78/0.62/0.69	0.79/0.56/0.66	0.78/0.58/0.67	0.80
adv + SC + MTurk + w-dist	0.81/0.60/0.68	0.76/0.54/0.63	0.78/0.56/0.65	0.77
att+adv + SC + MTurk + w-dist	0.84/0.66/0.73	0.81/0.63/0.71	0.82/0.64/0.72	0.83
OPINE (as proposed, with beam-CRF)	0.84/0.72/0.77	**0.81/0.69/0.75**	**0.82/0.70/0.76**	**0.86**
OPINE (as proposed, with constr-CRF)	**0.84/0.73/0.78**	0.81/0.68/0.74	**0.82/0.70/0.76**	**0.86**

Table 1: OPINE vs. State-of-the-art: precision(P), recall(R), F1-score and semantic similarity (Sim.) on Stack Exchange data

Input Text Utterance	Intents
Is it possible to **navigate** back ... **to previous page** after save processing? ... I have a page where I click on a link and use navigateURL ... want to be able to go back to the previous calling page and **complete** the **processing** of the **save**...	navigate previous page, complete processing save
The "Your tweets retweeted" page ... **find** out all the users who **retweeted** a tweet of mine? ... have retweeted a tweet and what their **Twitter IDs** are?	find retweeted Twitter IDs
Is there a WordPress plugin that will **tweet when** a scheduled post is posted? ... will tweet when you **publish** a post, but none I have tried will do it on a **scheduled post**.	tweet when publish scheduled post
I'm starting a micro-school... I want to **manage sick notes** and **absences** ... How can I **synchronize** one **central** Google **Calendar** ... Parents should be able to schedule future absences and excuse past absences...	manage sick notes, manage absences, synchronize central calendar

Table 2: Effect of attention. Darker colored highlight shows a higher attention value. Boldface denotes presence of intent.

fully identified by an attention head. For instance, the second row shows the significance of *'find out', 'retweeted', 'tweet'* and *'what their Twitter IDs are'* for the intent *"find retweeted Twitter IDs"*. The attention heads are attentive to indicator cues likely to precede an actionable intent, such as *'possible to', 'want to be able to', 'how can I'* and *'I want to'*. Our attention mechanism captures the dependency between distant intent words, such as *'find'* and *'retweeted'* in the second row and *'publish'* and *'scheduled'* in the third row. It also associates the action *'manage'* in the last row with two objects.

Performance on SNIPS. We next discuss the performance of OPINE on utterances from the SNIPS NLU [Coucke *et al.*, 2018] intent detection benchmark dataset. As seen in Table 3, OPINE can drill down into high-level intent categories, to understand, summarize or hierarchically organize the specific fine-grained intents in them. An additional side benefit of discovering intents using OPINE is that it can identify relevant accompanying *slots* apart from the intents, without performing a dedicated slot filling task. For instance, in the *PlayMusic* category of SNIPSin Table 3, OPINE not only recognizes the basic intents of *'hear song'* or *'play album'*; but also the corresponding names of singers (e.g. *Leroi Moore, Eddie Vinson*), song albums (e.g. *Curtain Call*, *Concerto*), and music platforms (e.g. *Youtube*, *Zvooq*). Note that OPINE was trained on *out-of-domain* intent data (Stack Exchange), since we do not have ACTION-OBJECT annotations available for SNIPS. This represents a challenging zero-shot environment [Socher *et al.*, 2013] to examine OPINE's performance, where no information is available about the test data.

Intents discovered by OPINE in a SNIPS class	Intents discovered by OPINE in a SNIPS class
PlayMusic:	Search Creative Work:
hear Leroi Moore	*need movie times*
play Curtain Call Album	*find schedule Comedian*
listen youtube, hear rock genre	*see JLA adventures*
find concerto, open itunes	*check schedule BowTie cinemas*
play concerto Zvooq	*get movies neighborhood*
hear seventies track	*show schedule Rat Rod*
Search Screening Event:	Book Restaurant:
look show Vanity	*book reservation bar spa*
find saga Chump Change	*eat eastern european food*
looking Plant Ecology	*need table Quaryville*
get Elvis TV show	*book spot tea house*
locate Epic Picture	*book spot City Tavern*

Table 3: Some fine-grained intents discovered in the SNIPS dataset.

5 Conclusion

We introduce and address the novel problem of *Open Intent Discovery* via a sequence tagging approach, OPINE, in contrast to prior work of detecting intents via classification. OPINE harnesses a Bi-LSTM and CRF coupled with multi-headed self-attention and adversarial training. Extensive experiments on real-world data show substantial improvements of OPINE over competitive baselines. We also release a large collection of 25K intent-annotated instances from diverse domains. A detailed description of OPINE and an in-depth empirical analysis is available in the full version of our paper [Vedula *et al.*, 2020].

Acknowledgments

This work was supported by the National Science Foundation grant EAR-1520870, the Ohio Supercomputer Center [Center, 1987] and Adobe Research.

References

[Angeli *et al.*, 2015] Gabor Angeli, Melvin Jose Johnson Premkumar, and Christopher D Manning. Leveraging linguistic structure for open domain information extraction. In *Proceedings of ACL-IJCNLP*, pages 344–354, 2015.

[Center, 1987] Ohio Supercomputer Center. Ohio supercomputer center. http://osc.edu/ark:/19495/f5s1ph73, 1987. Accessed: 2020-01-01.

[Chen *et al.*, 2013] Zhiyuan Chen, Bing Liu, Meichun Hsu, Malu Castellanos, and Riddhiman Ghosh. Identifying intention posts in discussion forums. In *NAACL-HLT*, 2013.

[Coucke *et al.*, 2018] Alice Coucke, Alaa Saade, Adrien Ball, Théodore Bluche, et al. Snips voice platform: an embedded spoken language understanding system for private-by-design voice interfaces. *arXiv preprint arXiv:1805.10190*, 2018.

[Dahl *et al.*, 1994] Deborah A Dahl, Madeleine Bates, Michael Brown, William Fisher, et al. Expanding the scope of the atis task: The atis-3 corpus. In *Proceedings of the workshop on Human Language Technology*, 1994.

[Forney, 1973] G David Forney. The viterbi algorithm. *Proceedings of the IEEE*, 61(3), 1973.

[Goodfellow *et al.*, 2015] Ian J Goodfellow, Jonathon Shlens, and Christian E Szegedy. Explaining and harnessing adversarial examples. In *ICLR*, 2015.

[Graves *et al.*, 2013] Alex Graves, Abdel-rahman Mohamed, and Geoffrey Hinton. Speech recognition with deep recurrent neural networks. In *IEEE ICASSP*, 2013.

[Gupta *et al.*, 2014] Vineet Gupta, Devesh Varshney, Harsh Jhamtani, Deepam Kedia, and Shweta Karwa. Identifying purchase intent from social posts. In *ICWSM*, 2014.

[Hochreiter and Schmidhuber, 1997] Sepp Hochreiter and Jürgen Schmidhuber. Long short-term memory. *Neural computation*, 1997.

[Huang *et al.*, 2015] Zhiheng Huang, Wei Xu, and Kai Yu. Bidirectional lstm-crf models for sequence tagging. *arXiv preprint arXiv:1508.01991*, 2015.

[Kim and Kim, 2018] Joo-Kyung Kim and Young-Bum Kim. Joint learning of domain classification and out-of-domain detection with dynamic class weighting for satisficing false acceptance rates. *arXiv preprint arXiv:1807.00072*, 2018.

[Kim *et al.*, 2016] Joo-Kyung Kim, Gokhan Tur, Asli Celikyilmaz, Bin Cao, and Ye-Yi Wang. Intent detection using semantically enriched word embeddings. In *IEEE SLT*, 2016.

[Kim *et al.*, 2017] Young-Bum Kim, Sungjin Lee, and Karl Stratos. Onenet: Joint domain, intent, slot prediction for spoken language understanding. In *ASRU workshop*, 2017.

[Lafferty *et al.*, 2001] John Lafferty, Andrew McCallum, and Fernando Pereira. Conditional random fields: Probabilistic models for segmenting and labeling sequence data. In *ICML*, 2001.

[Lin and Xu, 2019] Ting-En Lin and Hua Xu. Deep unknown intent detection with margin loss. *arXiv preprint arXiv:1906.00434*, 2019.

[Liu and Lane, 2016] Bing Liu and Ian Lane. Attention-based recurrent neural network models for joint intent detection and slot filling. *arXiv preprint arXiv:1609.01454*, 2016.

[Manning *et al.*, 2014] Christopher Manning, Mihai Surdeanu, John Bauer, Jenny Finkel, Steven Bethard, and David McClosky. The stanford corenlp natural language processing toolkit. In *Proceedings of ACL*, 2014.

[Miyato *et al.*, 2016] Takeru Miyato, Andrew Dai, and Ian Goodfellow. Adversarial training methods for semi-supervised text classification. *arXiv preprint arXiv:1605.07725*, 2016.

[Pennington *et al.*, 2014] Jeffrey Pennington, Richard Socher, and Christopher D. Manning. Glove: Global vectors for word representation. In *Proceedings of EMNLP*, 2014.

[Roth and Yih, 2005] Dan Roth and Wen-tau Yih. Integer linear programming inference for conditional random fields. In *Proceedings of ICML*, 2005.

[Socher *et al.*, 2013] Richard Socher, Milind Ganjoo, Christopher D Manning, and Andrew Ng. Zero-shot learning through cross-modal transfer. In *Advances in neural information processing systems*, pages 935–943, 2013.

[Srivastava *et al.*, 2015] Rupesh Kumar Srivastava, Klaus Greff, and Jürgen Schmidhuber. Highway networks. *arXiv preprint arXiv:1505.00387*, 2015.

[Tan *et al.*, 2018] Zhixing Tan, Mingxuan Wang, Jun Xie, Yidong Chen, and Xiaodong Shi. Deep semantic role labeling with self-attention. In *AAAI Conference on Artificial Intelligence*, 2018.

[Vaswani *et al.*, 2017] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, et al. Attention is all you need. In *NIPS*, 2017.

[Vedula *et al.*, 2020] Nikhita Vedula, Nedim Lipka, Pranav Maneriker, and S. Parthasarathy. Open intent extraction from natural language interactions. *Proceedings of The Web Conference 2020*, 2020.

[Wang *et al.*, 2015] Jinpeng Wang, Gao Cong, Wayne Xin Zhao, and Xiaoming Li. Mining user intents in twitter: A semi-supervised approach to inferring intent categories for tweets. In *AAAI*, 2015.

[Xia *et al.*, 2018] Congying Xia, Chenwei Zhang, Xiaohui Yan, Yi Chang, and Philip S Yu. Zero-shot user intent detection via capsule neural networks. *arXiv preprint arXiv:1809.00385*, 2018.

[Zhang and Wang, 2016] Xiaodong Zhang and Houfeng Wang. A joint model of intent determination and slot filling for spoken language understanding. In *IJCAI*, 2016.