Learning OOV through Semantic Relatedness in Spoken Dialog Systems

Ming Sun, Yun-Nung Chen, and Alexander I. Rudnicky

School of Computer Science, Carnegie Mellon University
5000 Forbes Avenue, Pittsburgh, PA 15213-3891, USA
{mings, yvchen, air}@cs.cmu.edu

Abstract

Ensuring language coverage in dialog systems can be a challenge, since the language in a domain may drift over time, creating a mismatch between the original training data and current input. This in turn degrades performance by increasing misunderstanding and eventually leading to task failure. Without the capability of adapting the vocabulary and the language model based on certain domains or users, recognition errors may degrade the understanding performance, and even lead to a task failure, which incurs more time and effort to recover. This paper investigates how coverage can be maintained by automatically acquiring potential out-of-vocabulary (OOV) words by leveraging different types of relatedness between vocabulary items and words retrieved from web-based resources. Our experiments show that both recognition and semantic parsing accuracy can thereby be improved.

Index Terms: speech recognition, spoken dialog system (SDS), OOV learning, word embeddings.

1. Introduction

Most speech recognition systems are closed-vocabulary and do not accommodate out-of-vocabulary (OOV) words. However, a lot of applications such as voice search or spoken dialog systems face the challenge that OOV words are usually content words such as locations and movie names, which carry the crucial information for the task success. Therefore, a domain-specific lexicon plays a crucial role on successful performance in spoken dialog systems. For example, a dialog system in a movie domain should have a corresponding vocabulary and a language model to get better system performance (including both recognition and understanding). However, the vocabulary is often fixed and determined prior to deployment [1, 2, 3, 4], which limits what language a system can understand. Even with the large vocabulary speech recognition, such as cloud-based ASR, the size of vocabulary is still finite. Inevitably, a dialog system with static vocabulary has to face the OOV issue after deployment, e.g., with the newly created words such as "selfie". Moreover, when an OOV occurs, the misrecognition affects not only the target OOV word but also words around it [5]. Even with a cloud ASR, to improve the recognition accuracy, a domain vocabulary should be considered to recover reliable transcriptions [6]. Hence, the issues can be addressed by introducing the adaptable vocabulary and language model for including domain knowledge [7].

To expand the domain-specific information, a domain vocabulary and a domain language model are required. There are several challenges about domain-specific language: 1) The building process may not be unsupervised, which requires the specified domain knowledge [7]. 2) Vocabulary expansion brings pros and cons to recognition, improving the word coverage but introducing acoustic confusions [5]. The goal of this work is to acquire potential OOVs in an unsupervised manner and balance the trade-off of vocabulary expansion.

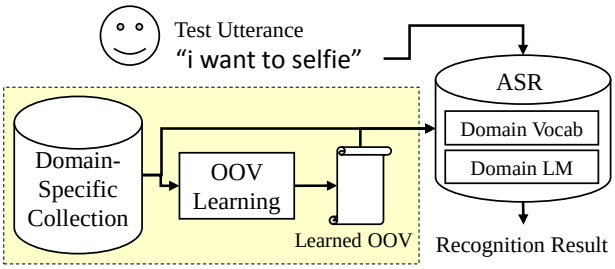

Figure 1: The *expect-and-learn* framework

A lot of work have focused on detecting and learning OOVs during human-machine conversations. OOV detection has been tackled from various perspectives: word/phone alignment [8, 9], classification [10, 11], or explicitly representing OOVs with fragments in decoding[12, 13]. Learning is usually done by performing phoneme-grapheme alignment [12, 14]. However, this *detect-and-learn* approach can only learn a limited number of new words discovered in the observed conversations. Moreover, to achieve reliable learning, it usually costs further dialog turns and human effort. In stead of learning OOVs after detecting them, this paper proposes to use a data-driven and knowledge-driven approach, *expect-and-learn*, which utilizes semantic resources to automatically enrich recognition vocabulary and the language model beforehand. The learned OOVs are more likely to be seen in the testing data because they are semantically related to the limited domain-specific data. Theoretically, the number of new words that can be generated is adaptable. The need for the help from human users is not required either.

This approach is inspired by the work in [15], which manually adds synonyms of in-vocabulary (IV) words to improve the system performance. Recent works utilized semantic similarity to project the input text (with out-of-grammar words) into task spaces (grammar concepts) for better language understanding [16, 17, 18, 19, 20, 21], but they did not include such information for recognition models. Considering automatic speech recognition (ASR) is intuitively more vulnerable in the presence of OOVs, our work focuses on proactively improving the coverage of the recognition vocabulary, taking broader word relatedness into consideration. As a result, both recognition and language understanding (semantic parsing) can be improved.

Figure 1 shows the proposed *expect-and-learn* framework. First based on the limited domain-specific data, an OOV learning procedure is applied to generate a list of OOVs that may be

domain-specific. The vocabulary and the language model can be expanded to cover more domain-related OOVs beforehand, resulting in better recognition and understanding performance without more conversations from users. In the following sections, the proposed OOV learning procedures are described in Section 2. Section 3 evaluates and discusses the performance, and Section 4 concludes.

2. OOV Learning Procedure

To learn OOVs based on the small domain-specific training data $\mathcal{D}$, this section considers measuring word relatedness through different resources, and then extracts OOVs that may be more likely to be observed in testing data. The learned OOVs are included into speech recognition vocabulary and language model before testing data comes in, which is the *expect-and-learn* strategy. Note that the *detect-and-learn* strategy expands the language coverage during recognition, where OOVs are recovered after first-pass recognition.

2.1. Relatedness Measurement

The word semantic relatedness can be obtained by two ways: 1) linguistically semantic relatedness and 2) data-driven semantic relatedness. First, we define a set of words as V, which includes the top N frequentest IV words in the domain-specific training data $\mathcal{D}$. Given an IV word set V, we propose following methods to generate a set of OOV candidates W and a word relatedness matrix M.

2.1.1. Linguistically Semantic Relatedness

The word semantic relatedness is defined by linguists under the assumption that words that have similar senses in common are more related to each other. For example, (*knock, punch*) is more related to each other than (*knock, kick*), since "*knock*" and "*punch*" use hand to touch an object while "*kick*" uses foot. WordNet [22] is a dictionary including such information. The word relatedness between a word pair can be measured by the LCH similarity metric [22, 23], which on average best correlates with human judgment, requiring no additional domain corpus [24].

Ganitkeyvitch et al. also developed a semantic relatedness database for paraphrasing (PPDB), where the words in each related word-pair can be translated into the same foreign word [25]. Here we use the PPDB-L (large size), which has better balance between coverage and accuracy. The similarity between a word pair (x, y) is measured by

$$Sim(x, y) = p(x \mid y) \sim \sum_{f} p(x \mid f)p(f \mid y), \qquad (1)$$

where $p(x \mid y)$ is the conditional paraphrase probability by marginalizing over all shared foreign-language translations f [25].

2.1.2. Data-Driven Semantic Relatedness

In recent, data-driven knowledge is utilized according to distributional semantics [16, 19]. Here we assume that words occurring in proximity can also be related. For example, (*knock, door*) is more related than (*knock, floor*) since the former occurs more often. Also, words sharing common contexts are more related. For example, (*cat, dog*) is more related than (*cat, turtle*) in the context of "*__ is running in the room*", because 1) both "*cat*" and "*dog*" are animals that move fast; and 2)"*cat*" and "*dog*" are more often observed as "*running*".

To involve OOV candidates based on the distributional semantics, we leverage the external data to train word embeddings by a continuous bag-of-word (CBOW) architecture to represent each word as a continuous-valued vector[1]. Here a large off-the-shelf model (300 dimension word vectors trained on 100 billion Google News words) is applied [26]. Then the similarity between words can be measured as the cosine similarity between their word embedding vectors. Under the assumption of distributional semantics, higher similarity suggests that the word pair occurs together more frequently.

2.2. Learning Algorithm

Given an IV set V and the similarity measures, we can build a set of OOV candidates W including the top N similar words to any IV word $v \in V$ based on the linguistic resources. Considering that web resources often have noisy information, data-driven semantic relatedness introduced in Section 2.1.2 may generate some noisy words in W. This issue is addressed by filtering out words that are more likely to be noises. Here when building the OOV set W, we only keep the word w with the frequency higher than T in a large external data to remove possible noises.

Also, a word relatedness matrix M is built based on the semantic relatedness introduced above. The entry of this matrix is the similarity between an IV word and an OOV word: $M(i, j) = Sim(v_i, w_j)$, where $v_i \in V, w_j \in W$. Below we simplify the notation $M_{x,y}$ as the entry corresponding to the similarity between the word pair (x, y). Two learning algorithms are proposed as follows.

2.2.1. Algorithm 1: Local OOV Learning Procedure

This algorithm learns the OOV words based on the most frequent IV words, where we iteratively extract the most related OOV word w^* for each IV word. The assumption is that, for each IV word, only the OOV word with highest semantic relatedness is reliable enough to be domain-specific OOV.

Algorithm 1 Local OOV Learning Procedure

Require: a set IV words V; a set of OOV candidates W, the word relatedness matrix M, a frequency function $f_{\mathcal{D}}(v)$ indicating the word frequency v in domain-specific data $\mathcal{D}$;
Ensure: a set of newly-learned OOV words $W^* \subset W$
1: Initializing $W^* = \{\}, V^* = \{\}$;
2: **repeat**
3: Deciding a most frequent IV word from the IV set, $v^* = \arg\max_{v \in \{V - V^*\}} f_{\mathcal{D}}(v)$;
4: Extracting a most prominent OOV word from the OOV candidate set, $w^* = \arg\max_{w \in \{W - W^*\}} M_{w,v^*}$;
5: Updating processed sets $W^* = W^* + w^*$ and $V^* = V^* + v^*$
6: **until** $|W^*| > \theta$
7: **return** W^*;

2.2.2. Algorithm 2: Global OOV Learning Procedure

The algorithm learns an OOV subset W^* that has the highest relatedness to the whole IV set V instead of individual IV words.

[1] https://code.google.com/p/word2vec/

The learned W^* satisfies

$$W^* = \arg\max_{W'} \sum_{w \in W'} M_w \cdot \boldsymbol{f_D}^T, \quad (2)$$

where $\boldsymbol{f_D} = [f_{\mathcal{D}}(v_1), ..., f_{\mathcal{D}}(v_{|V|})]^T$ to weight the prominence by the IV word frequency. Therefore, the learned OOVs consider more global relatedness comparing to Algorithm 1. Here the assumption is that an OOV that is more related to the whole IV set should be more important and domain-specific. The optimal subset can be learned via a greedy algorithm shown below.

Algorithm 2 Global OOV Learning Procedure

Require: a set IV words V; a set of OOV candidates W, a word relatedness matrix M, a frequency vector $\boldsymbol{f_D}$;
Ensure: a set of newly-learned OOV words $W^* \subset W$
1: Initializing $W^* = \{\}, V^* = \{\}$;
2: **repeat**
3: Extracting a most prominent OOV word based on the whole IV set $w^* = \arg\max_w M_w \cdot \boldsymbol{f_D}$, where $w^* \in \{W - W^*\}$;
4: Updating the processed set $W^* = W^* + w^*$
5: **until** $|W^*| > \theta$
6: **return** W^*;

2.3. Language Expansion

By the two algorithms above, we can decide a size of OOV set θ and obtain a learned OOV list W^*, where the words in W^* is more likely to carry important domain-specific information, since it is learned from a domain-specific IV set. The vocabulary can be expanded by adding the learned OOVs into it. In addition to expanding the vocabulary, the corresponding language model should be updated to incorporate the newly-learned OOVs, where Kneser-Ney smoothing technique is applied to better estimate the probabilities of these new-learned unigrams. With the expanded vocabulary and language model, we perform decoding by using the same acoustic model to evaluate the recognition performance, and furthermore, the semantic parsing is performed to test the understanding performance.

3. Experiments

3.1. Experimental Setup

To demonstrate performance of our proposed OOV learning method, we examine the results on the Wall Street Journal (WSJ) dataset. Since dialog systems are often constrained by the vocabulary size and available training data, we use the same size of data for training and testing, where the numbers of training, testing, and development sentences are 546, 546, 300 respectively. Here the dev set is for tuning the parameters including the filtering threshold T, and so on. We adopted standard WSJ GMM-HMM semi-continuous acoustic model to avoid the influence of other factors. Pronunciations for the learned OOVs are automatically generated by CMU dictionary[2] and LOGIOS Lexicon Tool[3].

[2] http://www.speech.cs.cmu.edu/cgi-bin/cmudict

[3] http://www.speech.cs.cmu.edu/tools/lextool.html

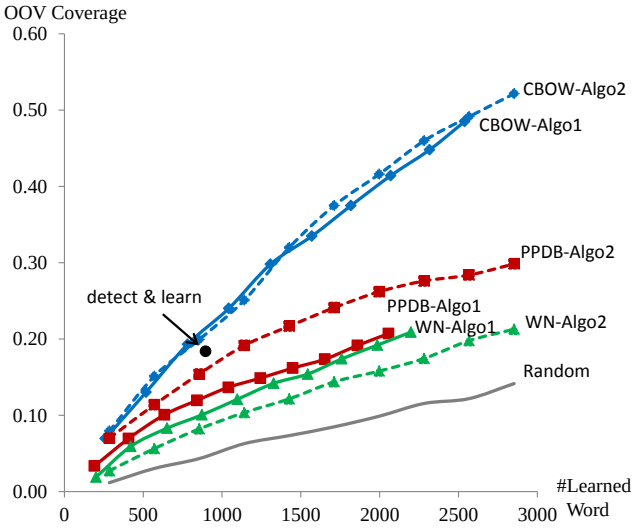

Figure 2: The OOV prediction performance across different resources (CBOW: data-driven continuous bag-of-words, WN: WordNet, PPDB: paraphrase database)

3.2. OOV Coverage of Resources

To compare different learning procedures using different resources, WordNet and PPDB for linguistic resources and CBOW word vectors for data-driven relatedness, we measure how many percentages of OOV tokens in the test set can be covered by the learned new words to evaluate the OOV learning quality and effectiveness of each resource. The baseline randomly chooses words from the generic dictionary (US English generic language model[4]) for vocabulary expansion. The results are shown in Figure 2.

It is shown that all methods are better than the baseline (gray line without markers), which demonstrates that all types of word relatedness in this paper can be used to effectively predict new words. Among all results using the local OOV learning procedure (Algorithm 1), the data-driven semantic relatedness (CBOW) outperforms others, while PPDB and WordNet do not show significant difference. Among all results using global OOV learning procedure (Algorithm 2), CBOW also performs best. The possible reason is that people have predictable language structure in their mind and just replace some words to form new sentences, so the data-driven technique is able to estimate better word relatedness when considering more complete language structures such as whole sentences instead of only words or phrases themselves. Comparing between two learning algorithms shows that the global OOV learning procedure produces better performance for CBOW and PPDB, since it considers the relatedness to the whole IV set instead of individual IV words. However, the results of WordNet perform differently, probably because WordNet is only good at measuring closely-related word pairs, and then the global consideration degrades the learning quality.

All proposed learning procedures are *expect-and-learn*, while another strategy, *detect-and-learn*, is shown as a dark point in the figure. It refers to the performance of OOV coverage in test set by adding all OOV words in dev set. It is shown that *detect-and-learn* is better than PPDB and WordNet

[4] http://sourceforge.net/projects/cmusphinx/files

Table 1: The recognition and understanding performance of OOV learning procedures

Vocab & LM	OOV Learning		Vocab Size	OOV Rate (%)	Recognition	Understanding		
					WER (%)	P (%)	R (%)	F (%)
Domain-Specific	Before	(a) Baseline	2854	22.6	49.9	62.6	52.3	57.0
	After	(b) Algo 1	5394	11.7	41.6	62.4	68.7	65.4
		(c) Algo 2	5394	11.6	42.0	61.8	68.8	65.1
		(d) Oracle	4254	0.0	23.5	81.4	80.5	80.9
Generic + Domain	Before	(e) Baseline	20175	3.6	21.7	80.2	84.4	82.2
	After	(f) Algo 1	22599	3.0	20.3	81.7	84.8	83.2
		(g) Algo 2	22599	3.0	20.4	81.6	84.9	83.2
		(h) Oracle	20431	0.0	15.1	86.9	87.3	87.1

but worse than CBOW. The better performance of our proposed algorithms and unlimited number of OOVs show the feasibility of improving the system performance through the *expect-and-learn* strategy.

3.3. Recognition Results

Table 1 shows the word error rate (WER) performance before and after performing the proposed OOV learning procedures. For rows (a)-(d), we only use the domain-specific training data $\mathcal{D}$ to build the vocabulary and the language model. Row (a) is the baseline result, which only takes domain training data for model training, and performs poor recognition results due to limited domain-specific training data. Rows (b) and (c) apply Algorithm 1 and 2 to learn OOVs through the data-driven semantic relatedness respectively (the best among different resources discussed in Section 3.2). It is shown that after learning OOVs by the proposed algorithms, the OOV rates significantly decrease, and recognition performance is also improved in both algorithms. Comparing between two algorithms, their performance is close to each other, which aligns well with the finding from Figure 2. To examine the potential of the OOV learning technique, row (d) shows the oracle results by adding all OOVs in testing data into domain vocabulary and language model. The performance can be referred as the upper bound, where the WER can be decreased from 50% to around 24%, showing the promising potential of OOV learning techniques.

We interpolate the US English generic language model with the domain language model (augmented by the acquired OOVs) to analyze the effectiveness of the proposed approaches, shown in rows (e)-(h). Similarly, it is found that applying the OOV learning approaches improves the recognition performance compared with the baseline where the US English model is used. Also, the oracle result (row (h)) still shows the potential room of improvement. In Generic + Domain condition, the US English vocabulary (about 20K words) already covers most of the words in test set (yielding only 3.6% OOV rate in the baseline). Our learned OOVs, which are outside the generic vocabulary, still captures useful OOVs. As a result, recognition performance is further improved. We believe in a more mismatched situation where dialog system developers have to deal with limited domain data, together with a mismatched generic model, the improvement would be more noticeable.

3.4. Language Understanding Results

In addition to recognition performance, we also examine the understanding performance after learning OOVs. The reason is that the recognition would be better if we successfully learn some words that are not really important for dialog systems. To evaluate the understanding performance, we perform semantic parsing on all utterances and extract the outputted semantic frames by SEMAFOR, a state-of-the-art frame semantic parser [27]. The reference semantic frames are outputted by the parser using the manual transcribed sentences. By comparing the outputted semantic frames from manual transcripts and decoded results, precision, recall, and F-measure are reported to evaluate the understanding performance, which are shown in the last three columns of Table 1.

It is obvious that understanding performance may be effected by OOVs. For rows (b) and (c), where a dialog system is built with limited domain data, the understanding performance after learning OOVs becomes better (from 57% to 65% on F-measure). The oracle performance achieves even 81% on F-measure, showing that it is very important for a system to adapt its vocabulary so as to ensure a reasonable language understanding performance. The similar conditions can be found in the rows (e)-(h). They both suggest that the OOV learning procedure is important in dialog systems, since the misunderstanding usually results in task failures. The proposed data-driven OOV learning procedures show the potential and the feasibility of improving dialog system performance through the *expect-and-learn* strategy.

4. Conclusion

This paper shows that speech recognition and language understanding performance can be improved through an OOV learning procedure. It is found that a limited domain vocabulary can be utilized to effectively acquire OOVs by the word relatedness theory through the use of web knowledge bases. With data-driven semantic relatedness, both the global and local learning procedures are able to successfully harvest more than 50% of OOVs, leading to better recognition and understanding performance. To summarize, the main contribution of this work is to demonstrate that OOV learning may benefit spoken dialog system and the proposed *expect-and-learn* strategy outperforms the traditional *detect-and-learn* in both higher effectiveness and no human involvement.

5. References

[1] G. W. Furnas, T. K. Landauer, L. M. Gomez, and S. T. Dumais, "The vocabulary problem in human-system communication," *Communications of the ACM*, vol. 30, no. 11, pp. 964–971, 1987.

[2] G. Chung, S. Seneff, C. Wang, and L. Hetherington, "A dynamic vocabulary spoken dialogue interface," in *Proc. ICSLP*, 2004, pp. 1457–1460.

[3] H. Holzapfel, D. Neubig, and A. Waibel, "A dialogue approach to learning object descriptions and semantic categories," *Robotics and Autonomous Systems*, vol. 56, no. 11, pp. 1004–1013, 2008.

[4] V. W. Zue and J. R. Glass, "Conversational interfaces: Advances and challenges," *Proceedings of the IEEE*, vol. 88, no. 8, pp. 1166–1180, 2000.

[5] R. Rosenfield, "Optimizing lexical and ngram coverage via judicious use of linguistic data," 1995.

[6] J. Twiefel, T. Baumann, S. Heinrich, and S. Wermter, "Improving domain-independent cloud-based speech recognition with domain-dependent phonetic post-processing," in *Twenty-Eighth AAAI Conference on Artificial Intelligence. Québec City, Canada*, 2014.

[7] L. Qin and A. Rudnicky, "Building a vocabulary self-learning speech recognition system," in *Fifteenth Annual Conference of the International Speech Communication Association*, 2014.

[8] S. Hayamizu, K. Itou, and K. Tanaka, "Detection of unknown words in large vocabulary speech recognition," in *Journal of the Acoustical Society of Japan (E) 16*, pp. 165–171.

[9] C. White, G. Zweig, L. Burget, P. Schwarz, and H. Hermansky, "Confidence estimation, OOV detection and language id using phone-to-word transduction and phone-level alignments," in *ICASSP*, 2008.

[10] H. Sun, G. Zhang, F. Zheng, and M. Xu, "Using word confidence measure for OOV words detection in a spontaneous spoken dialog system," in *Interspeech*, 2003.

[11] B. Lecouteux, G. Linares, and B. Favre, "Combined low level and high level features for out-of-vocabulary word detection," in *Interspeech*, pp. 1187–1190.

[12] M. Bisani and H. Ney, "Open vocabulary speech recognition with flat hybrid models," in *Interspeech*, pp. 725–728.

[13] L. Qin, M. Sun, and A. I. Rudnicky, "OOV detection and recovery using hybrid models with different fragments." in *INTERSPEECH*, 2011, pp. 1913–1916.

[14] K. Vertanen, "Combining open vocabulary recognition and word confusion networks," in *ICASSP*, pp. 4325–4328.

[15] W. Ward and S. Issar, "Recent improvements in the cmu spoken language understanding system," in *Proceedings of the workshop on Human Language Technology*. Association for Computational Linguistics, 1994, pp. 213–216.

[16] Y.-N. Chen and A. I. Rudnicky, "Dynamically supporting unexplored domains in conversational interactions by enriching semantics with neural word embeddings," in *Proceedings of SLT*, 2014.

[17] A. Pappu and A. I. Rudnicky, "Predicting tasks in goal-oriented spoken dialog systems using semantic knowledge bases," in *Proceedings of the SIGDIAL*, 2013, pp. 242–250.

[18] Y.-N. Chen, D. Hakkani-Tür, and G. Tur, "Deriving local relational surface forms from dependency-based entity embeddings for unsupervised spoken language understanding," in *Proceedings of SLT*, 2014.

[19] Y.-N. Chen, W. Y. Wang, and A. I. Rudnicky, "Leveraging frame semantics and distributional semantics for unsupervised semantic slot induction in spoken dialogue systems," in *Proceedings of SLT*, 2014.

[20] Y.-N. Chen, W. Y. Wang, A. Gershman, and A. I. Rudnicky, "Matrix factorization with knowledge graph propagation for unsupervised spoken language understanding," in *Proceedings of ACL-IJCNLP*, 2015.

[21] Y.-N. Chen, W. Y. Wang, and A. I. Rudnicky, "Jointly modeling inter-slot relations by random walk on knowledge graphs for unsupervised spoken language understanding," in *Proceedings of NAACL-HLT*, 2015, pp. 619–629.

[22] G. A. Miller, "Wordnet: a lexical database for english," *Communications of the ACM*, vol. 38, no. 11, pp. 39–41, 1995.

[23] C. Leacock and M. Chodorow, "Combining local context and wordnet similarity for word sense identification," *WordNet: An electronic lexical database*, vol. 49, no. 2, pp. 265–283, 1998.

[24] A. Budanitsky and G. Hirst, "Semantic distance in wordnet: An experimental, application-oriented evaluation of five measures," in *Workshop on WordNet and Other Lexical Resources*, vol. 2, 2001.

[25] J. Ganitkevitch, B. Van Durme, and C. Callison-Burch, "PPDB: The paraphrase database." in *HLT-NAACL*, 2013, pp. 758–764.

[26] T. Mikolov, I. Sutskever, K. Chen, G. S. Corrado, and J. Dean, "Distributed representations of words and phrases and their compositionality," in *Advances in Neural Information Processing Systems*, 2013, pp. 3111–3119.

[27] D. Das, D. Chen, A. F. Martins, N. Schneider, and N. A. Smith, "Frame-semantic parsing," *Computational Linguistics*, vol. 40, no. 1, pp. 9–56, 2014.