Dependency Parsing for Spoken Dialog Systems

Sam Davidson[1], Dian Yu[2], Zhou Yu[2]
[1]Department of Linguistics
[2]Department of Computer Science
University of California, Davis
Davis, CA 95616, USA
{ssdavidson,dianyu,joyu}@ucdavis.edu

Abstract

Dependency parsing of conversational input can play an important role in language understanding for dialog systems by identifying the relationships between entities extracted from user utterances. Additionally, effective dependency parsing can elucidate differences in language structure and usage for discourse analysis of human-human versus human-machine dialogs. However, models trained on datasets based on news articles and web data do not perform well on spoken human-machine dialog, and currently available annotation schemes do not adapt well to dialog data. Therefore, we propose the Spoken Conversation Universal Dependencies (SCUD) annotation scheme that extends the Universal Dependencies (UD) (Nivre et al., 2016) guidelines to spoken human-machine dialogs. We also provide ConvBank, a conversation dataset between humans and an open-domain conversational dialog system with SCUD annotation. Finally, to demonstrate the utility of the dataset, we train a dependency parser on the ConvBank dataset. We demonstrate that by pre-training a dependency parser on a set of larger public datasets and fine-tuning on ConvBank data, we achieved the best result, 85.05% unlabeled and 77.82% labeled attachment accuracy.

1 Introduction

Syntactic parsing (Chen et al., 2018) and semantic parsing (Kollar et al., 2018; Gupta et al., 2018) have been used in dialog systems to disambiguate the relationships between noun phrases in understanding tasks. Compared to constituency parsing and semantic role labeling, dependency parsing provides more clear relationships between predicates and arguments (Johansson and Nugues, 2008). Constituency parsers provide information about noun phrases in a sentence, but provide only limited information about relationships within a noun phrase. For example, in the sentence "What do you think about Google's privacy policy being reviewed by journalists from CNN?," a constituency parser would place "Google's privacy policy being reviewed by journalists from CNN" under a single phrasal node. Similarly, a semantic role labeling system would tend to label the same phrase as an argument of the verb, but it would not disambiguate the relationships within the phrase. Finally, NER only provides information about named entities which may or may not be the key semantic content of the sentence. Dependency parsers, by contrast, can provide information about relationships when a sentence contains multiple entities, even when those entities are within the same phrase.

Identifying relationships between entities in a user utterance can help a dialog system formulate a more appropriate response. For instance, in the sentence about "Google's privacy policy" mentioned above, there are multiple entities for the system to consider. The system must determine the most important entity in the utterance in order to model the topic and generate an appropriate response. A dependency graph facilitates this process by providing information about the syntactic relationships between entities.

Figure 1: Example dependency-parsed sentence. Stanford CoreNLP visualizer (Manning et al., 2014).

A dependency parse of the sentence, shown in Figure 1, demonstrates that the key element of the sentence, "policy", is directly linked to the root "think" as a *nmod*. The other entities in the sentence are modifiers of the central element, and are clearly labeled as such. These structures make it

easier for a dialog system to identify core semantic content and formulate an appropriate response.

Dependency parsing is a well studied (Dozat and Manning, 2016; Clark et al., 2018) task. However, most dependency parsers are trained on well-written text datasets, such as news articles and web-scrape data. In a conversational setting, dialog systems deal with transcripts that are produced by automated speech recognition (ASR) systems. ASR systems are prone to produce errors, such as dropped and mis-transcribed words, and prematurely terminated utterances. Such errors, in our experience, make building an automated dependency parser harder.

Additional data is needed to train parsers for the dialog system domain. Due to limitations in the UD scheme for parsing speech data, we propose an extension of the UD standard, SCUD, designed specifically for dialog systems. SCUD considers unique issues in automated speech transcription and what is useful for dialog systems to generate appropriate responses. We then use SCUD to annotate a corpus of open-domain human-system conversations with automated speech transcripts. We finally leverage this data, along with previously annotated UD datasets, to build a dependency parser. Since the training data used is collected with an open-domain dialog system, the trained dependency parser can be applied to dialog systems with any task.

In addition to its use in dialog systems, dependency parsed dialog data could be used in discourse analysis to assist in understanding the syntactic patterns people use when interacting with a computer. While researchers have investigated such differences in task-oriented dialog systems (Doran et al., 2003), we believe that ConvBank and SCUD present an opportunity for expanded discourse analysis of open-domain dialog systems.

2 Related Work

Dependency parsing on text such as the Universal Dependencies English Web Treebank (UD-EWT) (Silveira et al., 2014), is a well-developed field in NLP; however, the parsing of automated speech transcripts remains an open problem (Bechet et al., 2014). As has been well-documented, speech, and the resulting ASR transcripts, is a very different domain from written text (Biber and Conrad, 1999; Leech, 2000; Carter and McCarthy, 2017). Adams (2017) focuses on adapting a parser trained on

UD-EWT to the speech domain without annotating new training data. For example, she attempts to create additional speech domain training data by parsing raw speech transcripts with Google's SyntaxNet. She then uses the resulting parsed output as training data for MaltParser (Nivre et al., 2006). However, the resulting models do not show consistent improvement over a baseline parser. Based on these results, and the apparent need for new training data for the speech domain, we chose to annotate a new human-system dialogs dataset to facilitate building dependency parsers for automated speech transcription and dialog systems.

Previous work has demonstrated that expanding the UD annotation scheme can result in successful parsers for other domains. For example, Liu et al. (2018) expand the UD scheme to train a parser for Twitter data. Thus we take a similar approach in expanding the UD scheme to encompass issues common to automated speech transcription.

Unlike written discourse, speech is full of disfluencies which make discovering the underlying syntactic structure challenging, as such disfluencies interrupt the syntactic structure of the utterance (Nasr et al., 2014; Caines et al., 2017). For example, according to Meteer and Taylor (1995), 17% of tokens in the Switchboard telephone conversations are various disfluencies. Frequent types of disfluencies include hesitations and false starts. When dealing with ASR-based systems, one must also contend with ASR errors. According to Nasr et al. (2014), such disfluencies must either be integrated into the syntactic parse of the utterance or removed prior to parsing. We choose to take the former approach, integrating speech disfluencies and speech transcription errors into the dependency graph produced by our annotation scheme and the resultant parser.

3 Annotation Scheme

When we initially began to annotate speech transcript data using the UD 2.0 (Nivre et al., 2017) scheme, our annotators repeatedly complained of constructions which they were unable to annotate given the limitations of UD 2.0. We developed SCUD in response to these issues. SCUD closely follows UD 2.0 so that we may leverage larger UD-compliant datasets of text data, such as UD-EWT. We extend the UD annotation scheme to encompass specific features peculiar to the speech domain, namely the use of non-syntactic discourse

markers, speech disfluencies, and ASR errors.

One major difference between speech and text is the fact that speech contains a fairly large number of discourse markers which do not serve a syntactic role in the utterance. This is similar to the difference between text and tweets pointed out by Liu et al. (2018). We make use of the UD relation *discourse* to indicate discourse markers which are not part of the syntactic structure of an utterance; for example, the word "like" in "I have like three dogs", where "like" plays a pragmatic, rather than a syntactic, role.

Another goal of our annotation scheme is to handle disfluencies common to both human and ASR-generated speech transcripts. We expand the scope of extant UD relation tags to tag common speech disfluencies. For example, the UD tag *reparandum* is applicable to situations in which a user self-corrects his or her speech. We use the UD tag *flat* to link repeated words and stutters to indicate that these headless multi-word expressions are a single syntactic unit. Handling non-syntactic tokens is particularly important because, while ASR may improve, the way people speak is less likely to change.

We are also able to expand the existing UD annotation scheme to handle some common errors in ASR output. In cases where an ASR system incorrectly splits words, we utilize the UD relation *goeswith* to link the two words in a dependency graph. We also provide a means to insert nodes at locations where an ASR system has dropped words necessary for formation of a grammatical structure. We do this by expanding the enhanced UD 2.0 standard, which provides a method to insert nodes in graphs as place-holders for words omitted through ellipsis. An example of using such a place-holder node is shown in Table 1, where the subject of the sentence in question has been omitted due to an ASR error. We also create one new relation tag to deal with a common error in ASR system output: prematurely terminated user utterances. Many conversational agents are turn-based (Kepuska and Bohouta, 2018) , and depend on pauses in user speech to detect the end of the user's dialog turn. As a result, hesitations by the user can result in the system prematurely terminating the user's turn. In reviewing our dialog system output data, we found that the ASR system prematurely terminates as many as 5% of user turns. We created the dependency label *preterm* to

Transcript: got two dogs				
Index	**Word**	**POS**	**Dep.**	**Relation**
1	E1.1	PRON	2	nsubj
2	got	VERB	0	root
3	two	NUM	4	nummod
4	dogs	NOUN	2	obj

Table 1: Example of omitted word node insertion

annotate such occurrences in our dataset.

We must also consider the issue of incorrectly transcribed words. ASR systems are prone to mistranscribing words in the input signal; such errors can significantly change the meaning of the underlying utterance or result in a transcription which is ungrammatical or semantically nonsensical (Errattahi et al., 2018). While error correction methods for dialog systems have been developed (Choi et al., 2016), preprocessing our data with such a system is beyond the scope of this study. We instructed annotators to annotate all words as they were transcribed by the system. The alternative would involve annotators trying to infer user intent from context, which makes predictable annotation far less likely.

In total, SCUD has 37 primary relation tags. The relation tagset closely follows the UD 2.0 annotation standards with the exception of domain-specific extensions as noted above.

4   Dataset and Annotation Process

We collected a corpus of humans conversing with an open-domain dialog chatbot based on the Alexa platform (Chen et al., 2018). Users were asked to come into a lab and have an undirected conversation with our system. The automated speech transcripts of the users' utterances were collected and selected at random for inclusion in this corpus. We recruited two annotators who are experienced with dependency grammars and annotation projects to annotate 1,500 sentences. As is common with other UD annotation projects, we asked each annotator to correct the work of the other. Once this round of corrections was complete, we measured agreement as the percentage of tokens which remained unchanged between the original annotations and the corrected version. The two annotators agreed on 86.4% of unlabeled dependencies and 80.5% of labeled dependencies. These numbers are comparable to those reported on Tweebank (Liu et al., 2018), another challeng-

UD-EWT		ConvBank	
Tag	**Freq.**	**Tag**	**Freq.**
punct	11.5%	root	15.1%
case	8.5%	nsubj	13.8%
nsubj	8.0%	obj	9.2%
det	7.7%	advmod	8.2%
root	6.1%	case	7.2%
advmod	5.4%	det	6.8%
obj	5.0%	obl	6.0%
obl	4.5%	aux	5.3%
amod	4.4%	cop	4.1%
compound	4.0%	amod	3.2%

Table 2: Comparison of common relation frequencies.

ing annotation problem. Once annotator agreement was assessed, a third annotator reviewed the annotations to improve compliance with the annotation scheme. We found that ASR errors, such as ungrammatical sentences resulting from mistranscription and word omission are a major contributing factor to annotator disagreement in our dataset.

The result of our annotation project is ConvBank, a corpus of 1,500 ASR-generated text utterances from a human-machine spoken dialog system with SCUD annotation. The corpus is compliant with UD 2.0 standards with the exception of extensions made to the UD standard as noted above. All data, code, and pre-trained models related to this project are publicly available via the project webpage[1].

ConvBank uses 32 of the 37 possible primary relations in SCUD, described above. A comparison of the ten most common tags in ConvBank and UD-EWT is presented in Table 2. *punct* is the most common relation in UD-EWT but is absent from ConvBank; this is because our automated speech transcripts do not contain punctuation. Other differences in relation distribution, such as the higher relative frequencies of *root* and *nsubj* can be explained by the shorter sentences in ConvBank compared to UD-EWT. Since each sentence must have a root, and most have a subject, the fewer other words in each sentence, the higher the frequencies of these two tags will tend to be.

5 Parsing Experiments

To demonstrate the utility of the ConvBank dataset, we use the data to train a dependency

parser for ASR-based spoken dialog systems.

5.1 Training Data

We use three datasets in training our dependency parser. We use the publicly available UD-EWT (Silveira et al., 2014) and Tweebank v.2 (Liu et al., 2018) to train baseline models. We also combine UD-EWT and Tweebank to create a larger UD dataset for pre-training. Finally, we use our own ConvBank dataset to fine-tune our best-performing baseline model. We hold out 500 annotated ConvBank sentences for testing.

We considered using the constituency parsed Switchboard corpus (Godfrey et al., 1992), which is a human-human telephone corpus, due to its similarity to our domain. However, no fully UD-compliant version of the dependency parsed corpus is publicly available. Although conversion from constituency to dependency graphs is possible, we found the resulting dependency graphs unusable due to the error-prone conversion process. Peng and Zeldes (2018) point out that conversion systems particularly struggle with attaching nominal modifiers and oblique arguments.

5.2 Methods

We build all models using the biaffine parser as described in Dozat and Manning (2016) and Qi et al. (2019). We use the open-source implementations of Dozat's biaffine parser code[2] for training our models. We use Glove (Pennington et al., 2014) for pre-trained word embeddings. We train three baseline models to compare how models trained on text data compare with our model, which leverages both annotated text and automated speech transcripts. First, we train a parsing model on UD-EWT (Silveira et al., 2014) alone. We also train a separate model on Tweebank v.2 (Liu et al., 2018) alone to determine how a parser trained on tweets, which are arguably more similar to conversational speech, will perform. We also train a model on combined UD-EWT and Tweebank data. Our final baseline is a model trained on a 1,000 utterance subset of ConvBank. This model is used to determine how a parser trained on our data performs without pre-training on a larger text dataset.

Our fine-tuning approach is based on the idea that, by providing the system with additional domain-specific examples from ConvBank, we are able to build upon the system's baseline knowl-

---

[1]https://gitlab.com/ucdavisnlp/dialog-parsing

[2]https://github.com/tdozat/Parser-v3

edge to create a final system which performs substantially better than baseline. Stymne et al. (2018) demonstrate that this type of fine-tuning is an effective way to leverage multiple treebanks to train a single dependency parser. Our fine-tuning process consists of initializing our network weights using the weights from our baseline model trained on the UD English Web Treebank. As demonstrated by our results, the baseline model trained on UD-EWT alone performs reasonably well in assigning many UD labels and dependencies; however the system does not handle well issues specific to the dialog domain. To improve performance on the dialog domain, we resume training the pre-trained model with our annotated ConvBank data, using identical training parameters, until perplexity stabilizes.

5.3 Results and Discussion

We present the parsing accuracy on the ConvBank test set in Table 3. Training only using 1,000 annotated in-domain ConvBank data reaches 75.65% and 58.29% accuracy on unlabeled and labeled edges respectively. This suggests that 1,000 utterances are not enough to achieve optimal results. However, human annotations are expensive to obtain, limiting the size of our dataset. The model trained on EWT or Tweebank or EWT and Tweebank combined achieved similar results that are better than the model trained on ConvBank alone. As these three models had similar results, we fine-tune on the slightly better-performing EWT model. Fine-tune achieves the best best result on both unlabeled and labeled edges (85.05% and 77.82% respectively). Fine-tuning on EWT with ConvBank data improves parsing accuracy on the test data by over 7 points on unlabeled dependency edges and by nearly 12 points on labeled edges. These results suggest leveraging both a large set of out-of-domain data and in-domain data achieves the best performance.

There are some clear differences between our fine-tuned model and the baseline model trained on text data. In speech, discourse markers such as "but", "so", and "and" frequently begin utterances. These items can be difficult to parse correctly, as they more frequently serve grammatical functions. Our baseline model generally tags such items as coordinating conjunctions or adverbs, while in the fine-tuned model such items are more often appropriately tagged as discourse markers. Another

Dataset	UAS	LAS
UD-EWT	77.41	65.85
Tweebank	77.23	66.44
Tweebank+UD-EWT	77.38	65.75
ConvBank	75.65	58.29
Fine-tune	85.05	77.82

Table 3: Parsing Accuracy Results

clear example is the correct use of the *reparandum* tag. For example in the utterance "you know you my name", the baseline model tags the second "you" as an indirect object, while the fine-tuned model correctly tags the item as *reparandum*. The baseline model tags only three items in our test data as *reparandum*, while the fine-tuned model identifies thirty-one such instances. These results indicate that our fine-tuned model is more closely aligned to the speech domain than our baseline model. However, our best model still struggles when faced with fixed expressions common in conversational speech, such as "Top Gun" and "Kitchen Table Wisdom". The model also performs poorly on ungrammatical utterances, as expected.

6 Conclusion

Dependency parsing can effectively identify the relationships between entities in a user utterance to improve speech language understanding. However, dependency parsing for speech data does not perform well when trained on out-of-domain text data. To solve this problem, we present ConvBank, a new dataset of annotated automated speech transcripts, annotated with the Spoken Conversation Universal Dependencies (SCUD) annotation scheme to build UD-compatible parses. We demonstrate the utility of ConvBank by fine-tuning a dependency parser with a large dataset, reaching 85.05% on labeled dependency edges and 77.82% on unlabeled dependencies.

In future work, we hope to expand the size of ConvBank to improve its utility to researchers in training parsing models and studying conversational systems. We also hope to improve the performance of our parsing system by jointly training dependency parsing with semantic role labeling.

References

Allison Adams. 2017. Dependency parsing and dialogue systems : an investigation of dependency parsing for commercial application.

Frederic Bechet, Alexis Nasr, and Benoit Favre. 2014. Adapting dependency parsing to spontaneous speech for open domain spoken language understanding. In *Fifteenth Annual Conference of the International Speech Communication Association*.

Douglas Biber and Susan Conrad. 1999. Lexical bundles in conversation and academic prose. *Language and Computers*, 26:181–190.

Andrew Caines, Michael McCarthy, and Paula Buttery. 2017. Parsing transcripts of speech. In *Proceedings of the Workshop on Speech-Centric Natural Language Processing*, pages 27–36.

Ronald Carter and Michael McCarthy. 2017. Spoken grammar: Where are we and where are we going? *Applied Linguistics*, 38(1):1–20.

Chun-Yen Chen, Dian Yu, Weiming Wen, Yi Mang Yang, Jiaping Zhang, Mingyang Zhou, Kevin Jesse, Austin Chau, Antara Bhowmick, Shreenath Iyer, et al. 2018. Gunrock: Building a human-like social bot by leveraging large scale real user data.

Junhwi Choi, Donghyeon Lee, Seounghan Ryu, Kyusong Lee, Kyungduk Kim, Hyungjong Noh, and Gary Geunbae Lee. 2016. *Engine-Independent ASR Error Management for Dialog Systems*, pages 193–203. Springer International Publishing, Cham.

Kevin Clark, Minh-Thang Luong, Christopher D. Manning, and Quoc V. Le. 2018. Semi-supervised sequence modeling with cross-view training. *CoRR*, abs/1809.08370.

Christine Doran, John Aberdeen, Laurie Damianos, and Lynette Hirschman. 2003. Comparing several aspects of human-computer and human-human dialogues. In *Current and new directions in discourse and dialogue*, pages 133–159. Springer.

Timothy Dozat and Christopher D Manning. 2016. Deep biaffine attention for neural dependency parsing. *arXiv preprint arXiv:1611.01734*.

Rahhal Errattahi, Salil Deena, Asmaa El Hannani, Hassan Ouahmane, and Thomas Hain. 2018. Improving asr error detection with rnnlm adaptation. In *2018 IEEE Spoken Language Technology Workshop (SLT)*, pages 190–196. IEEE.

John J. Godfrey, Edward C. Holliman, and Jane McDaniel. 1992. Switchboard: Telephone speech corpus for research and development. In *Proceedings of the 1992 IEEE International Conference on Acoustics, Speech and Signal Processing - Volume 1*, ICASSP'92, pages 517–520, Washington, DC, USA. IEEE Computer Society.

Sonal Gupta, Rushin Shah, Mrinal Mohit, Anuj Kumar, and Mike Lewis. 2018. Semantic parsing for task oriented dialog using hierarchical representations. *CoRR*, abs/1810.07942.

Richard Johansson and Pierre Nugues. 2008. Dependency-based semantic role labeling of propbank. pages 69–78.

Veton Kepuska and Gamal Bohouta. 2018. Next-generation of virtual personal assistants (microsoft cortana, apple siri, amazon alexa and google home). In *2018 IEEE 8th Annual Computing and Communication Workshop and Conference (CCWC)*, pages 99–103. IEEE.

Thomas Kollar, Danielle Berry, Lauren Stuart, Karolina Owczarzak, Tagyoung Chung, Lambert Mathias, Michael Kayser, Bradford Snow, and Spyros Matsoukas. 2018. The alexa meaning representation language. pages 177–184.

Geoffrey Leech. 2000. Grammars of spoken english: New outcomes of corpus-oriented research. *Language learning*, 50(4):675–724.

Yijia Liu, Yi Zhu, Wanxiang Che, Bing Qin, Nathan Schneider, and Noah A Smith. 2018. Parsing tweets into universal dependencies. *arXiv preprint arXiv:1804.08228*.

Christopher D. Manning, Mihai Surdeanu, John Bauer, Jenny Finkel, Steven J. Bethard, and David McClosky. 2014. The Stanford CoreNLP natural language processing toolkit. In *Association for Computational Linguistics (ACL) System Demonstrations*, pages 55–60.

Marie W. Meteer and Ann A. Taylor. 1995. Dysfluency annotation stylebook for the Switchboard corpus.

Alexis Nasr, Frederic Bechet, Benoit Favre, Thierry Bazillon, Jose Deulofeu, and Andre Valli. 2014. Automatically enriching spoken corpora with syntactic information for linguistic studies. In *LREC*, pages 854–858.

Joakim Nivre, Željko Agić, Lars Ahrenberg, Maria Jesus Aranzabe, Masayuki Asahara, Aitziber Atutxa, Miguel Ballesteros, John Bauer, Kepa Bengoetxea, Riyaz Ahmad Bhat, Eckhard Bick, Cristina Bosco, Gosse Bouma, Sam Bowman, Marie Candito, Gülşen Cebiroğlu Eryiğit, Giuseppe G. A. Celano, Fabricio Chalub, Jinho Choi, Çağrı Çöltekin, Miriam Connor, Elizabeth Davidson, Marie-Catherine de Marneffe, Valeria de Paiva, Arantza Diaz de Ilarraza, Kaja Dobrovoljc, Timothy Dozat, Kira Droganova, Puneet Dwivedi, Marhaba Eli, Tomaž Erjavec, Richárd Farkas, Jennifer Foster, Cláudia Freitas, Katarína Gajdošová, Daniel Galbraith, Marcos Garcia, Filip Ginter, Iakes Goenaga, Koldo Gojenola, Memduh Gökırmak, Yoav Goldberg, Xavier Gómez Guinovart, Berta Gonzáles Saavedra, Matias Grioni, Normunds Grūzītis, Bruno Guillaume, Nizar Habash, Jan

Hajič, Linh Hà Mỹ, Dag Haug, Barbora Hladká, Petter Hohle, Radu Ion, Elena Irimia, Anders Johannsen, Fredrik Jørgensen, Hüner Kaşıkara, Hiroshi Kanayama, Jenna Kanerva, Natalia Kotsyba, Simon Krek, Veronika Laippala, Phng Lê H`ông, Alessandro Lenci, Nikola Ljubešić, Olga Lyashevskaya, Teresa Lynn, Aibek Makazhanov, Christopher Manning, Cătălina Mărănduc, David Mareček, Héctor Martínez Alonso, André Martins, Jan Mašek, Yuji Matsumoto, Ryan McDonald, Anna Missilä, Verginica Mititelu, Yusuke Miyao, Simonetta Montemagni, Amir More, Shunsuke Mori, Bohdan Moskalevskyi, Kadri Muischnek, Nina Mustafina, Kaili Müürisep, Lng Nguy˜ên Thị, Huy`ên Nguy˜ên Thị Minh, Vitaly Nikolaev, Hanna Nurmi, Stina Ojala, Petya Osenova, Lilja Øvrelid, Elena Pascual, Marco Passarotti, Cenel-Augusto Perez, Guy Perrier, Slav Petrov, Jussi Piitulainen, Barbara Plank, Martin Popel, Lauma Pretkalniņa, Prokopis Prokopidis, Tiina Puolakainen, Sampo Pyysalo, Alexandre Rademaker, Loganathan Ramasamy, Livy Real, Laura Rituma, Rudolf Rosa, Shadi Saleh, Manuela Sanguinetti, Baiba Saulīte, Sebastian Schuster, Djamé Seddah, Wolfgang Seeker, Mojgan Seraji, Lena Shakurova, Mo Shen, Dmitry Sichinava, Natalia Silveira, Maria Simi, Radu Simionescu, Katalin Simkó, Mária Šimková, Kiril Simov, Aaron Smith, Alane Suhr, Umut Sulubacak, Zsolt Szántó, Dima Taji, Takaaki Tanaka, Reut Tsarfaty, Francis Tyers, Sumire Uematsu, Larraitz Uria, Gertjan van Noord, Viktor Varga, Veronika Vincze, Jonathan North Washington, Zdeněk Žabokrtský, Amir Zeldes, Daniel Zeman, and Hanzhi Zhu. 2017. Universal dependencies 2.0. LINDAT/CLARIN digital library at the Institute of Formal and Applied Linguistics (ÚFAL), Faculty of Mathematics and Physics, Charles University.

Joakim Nivre, Marie-Catherine De Marneffe, Filip Ginter, Yoav Goldberg, Jan Hajic, Christopher D Manning, Ryan T McDonald, Slav Petrov, Sampo Pyysalo, Natalia Silveira, et al. 2016. Universal dependencies v1: A multilingual treebank collection. In *LREC*.

Joakim Nivre, Johan Hall, and Jens Nilsson. 2006. Maltparser: A data-driven parser-generator for dependency parsing. In *In Proc. of LREC-2006*, pages 2216–2219.

Siyao Peng and Amir Zeldes. 2018. All roads lead to ud: Converting stanford and penn parses to english universal dependencies with multilayer annotations. In *Proceedings of the Joint Workshop on Linguistic Annotation, Multiword Expressions and Constructions (LAW-MWE-CxG-2018)*, pages 167–177.

Jeffrey Pennington, Richard Socher, and Christopher D. Manning. 2014. Glove: Global vectors for word representation. In *In EMNLP*.

Peng Qi, Timothy Dozat, Yuhao Zhang, and Christopher D Manning. 2019. Universal dependency parsing from scratch. *arXiv preprint arXiv:1901.10457*.

Natalia Silveira, Timothy Dozat, Marie-Catherine de Marneffe, Samuel Bowman, Miriam Connor, John Bauer, and Christopher D. Manning. 2014. A gold standard dependency corpus for English. In *Proceedings of the Ninth International Conference on Language Resources and Evaluation (LREC-2014)*.

Sara Stymne, Miryam de Lhoneux, Aaron Smith, and Joakim Nivre. 2018. Parser training with heterogeneous treebanks. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 619–625.