Towards Context-Aware Syntax Parsing and Tagging

Alaa Mohasseb
School of Computing, University of Portsmouth, United Kingdom
alaa.mohasseb@port.ac.uk

Mohamed Bader-El-Den
School of Computing, University of Portsmouth, United Kingdom
mohamed.bader@port.ac.uk

Mihaela Cocea
School of Computing, University of Portsmouth, United Kingdom
mihaela.cocea@port.ac.uk

Abstract

Information retrieval (IR) has become one of the most popular Natural Language Processing (NLP) applications. Part of speech (PoS) parsing and tagging plays an important role in IR systems. A broad range of PoS parsers and taggers tools have been proposed with the aim of helping to find a solution for the information retrieval problems, but most of these are tools based on generic NLP tags which do not capture domain-related information. In this research, we present a domain-specific parsing and tagging approach that uses not only generic PoS tags but also domain-specific PoS tags, grammatical rules, and domain knowledge. Experimental results show that our approach has a good level of accuracy when applying it to different domains.

2012 ACM Subject Classification Computing methodologies → Natural language processing

Keywords and phrases Information Retrieval, Natural Language Processing, PoS Tagging, PoS Parsing, Machine Learning

Digital Object Identifier 10.4230/OASIcs.ICCSW.2018.5

Category Main Track

1 Introduction

Parts-of-speech (PoS) tags play an important role in Natural Language Processing (NLP). PoS tagging provides a large amount of information about words. PoS parsing and tagging is one of the fundamental phases in text processing. Parsing has been used as a way to identify the sentence structure by adding mark-ups which helps in organizing a sentence, while tagging represent classes and features of words, in which each word will receive a tag based upon its word class and the feature it holds.

A broad range of PoS parsing and tagging tools and approaches have been developed; most of these tools and approaches are based on natural language. Furthermore, parsers and taggers still suffer from the problem of domain adaptation [21],[13] since most of them are based just on generic NLP tags which have a limited use in domains such as search engines, question answering systems and social networks; knowing only the generic PoS tags will not assist in identifying and retrieving relevant information since a lot of knowledge related to most of these domains cannot be captured with generic PoS tags. Moreover, most parser and tagger methods do not take inconsideration the syntax and grammatical structure of the given text.

2018 Imperial College Computing Student Workshop (ICCSW 2018).
Editors: Edoardo Pirovano and Eva Graversen; Article No. 5; pp. 5:1–5:9
OpenAccess Series in Informatics
OASICS Schloss Dagstuhl – Leibniz-Zentrum für Informatik, Dagstuhl Publishing, Germany

In this paper, we propose a Domain Specific Syntax-based Parsing and Tagging (DSSPT) approach. The aim of the research presented in this paper is to evaluate the influence of using domain-specific grammatical rules categories on the the parsing and tagging process and the classification performance. In addition, we aim to evaluate the use of DSSPT on two different domains: query classification and question classification.

The rest of the paper is organized as follows. Section 2 outlines previous work in parsing and tagging, including different proposed tools and approaches. Section 3 describes the proposed parsing and tagging framework. The experiments setup and results are presented in Section 4. Finally, Section 5 concludes the paper and outlines directions for future work.

2 Background

In this section we review previous work on parsing and tagging. Different methods of parsing are outlined in Section 2.1, while Section 2.2 reviews previous work on tagging methods.

2.1 Parsing

Many recent studies proposed different parsing methods and models; some of these are based on dependency parsing. Authors in [21] developed distant-supervised algorithms that use a dependency grammar for Community Question Answering (CQA). In [32] authors developed a graph-based and a transition-based dependency parser using beam-search, while in [8] a simple semi-supervised method for training dependency parsers was presented. Authors in [19] introduced MaltParser, a data-driven parser generator for dependency parsing. Some works used machine learning algorithms. Authors in [3] proposed a dependency parser using neural networks, while authors in [27] introduced algorithms to derive a query's syntactic structure from the dependency trees. Furthermore, in [26] authors proposed a general compositional vector framework for transition based dependency parsing. Other works introduced a semantic-based parser model. Works in [9] presented a semantic parsing model for answering compositional questions. Moreover, in [30] authors presented a statistical natural language semantic parsing modeling, while in [31] authors proposed a semantic parsing framework for question answering. Authors in [24] introduced a Compositional Vector Grammar (CVG), which combines probabilistic context-free grammar (PCFGs). In [11] authors proposed an algorithm of text parsing which was demonstrated on data from Twitter, while in [25] a recursive neural network architecture was introduced. Finally, authors in [28] proposed a technique for improving parser portability.

2.2 Tagging

Most taggers and tagging approaches have been developed for general PoS tagging. Authors in [20] proposed a tag-set that consists of twelve universal PoS categories. In [1] the authors proposed a Trigrams'n'Tags (TnT) statistical PoS tagger. Moreover, work in [4] proposed a PoS tagger based on Support Vector Machines (SVMT). Other works like [29] proposed a PoS tagger using dependency network representation. In [10] authors presented a method for unsupervised PoS tagging that considers a word type. Furthermore, few taggers have been developed for specific domains. In [5] authors addressed the problem of PoS tagging for English data from Twitter. In [7] a PoS tagging method for web search queries was proposed using the sentence level morphological analysis, while in [22] a probabilistic tagging method was proposed, which avoids the problems of Markov model based taggers. Finally, authors in [12] introduced an approach for deep parsing of web search queries using a context-free multiset generating grammar.

3 Proposed Approach

3.1 Tag-set

The tag-set was developed by [18] and updated by [17]. It was mainly created for the purpose of identifying search queries by labelling each word in the query with its PoS tag and name entity to help in the classification of the users' intent. The tag-set has been tested on different search engines' queries datasets [17], [15], i.e. AOL 2006 data-set[1] and the TREC 2009 Million Query Track data-set[2]. Furthermore, it has been used in other domains such as question classification [16] and also has been tested on different questions datasets, i.e. Yahoo Non-Factoid Question Dataset[3], TREC 2007 question answering data[4] and a Wikipedia dataset[5] that was generated by [23].

The tag-set consists of 10,440 different words that have been labelled with PoS tags (categories) which include three levels of details from our grammar taxonomy: (1) Level 1 includes the seven major word classes in English, which are *Verb (V), Noun (N), Determiner (D), Adjective (Adj), Adverb (Adv), Preposition (P)* and *Conjunction (Conj)* ; (2) Level 2 consists of sub-categories of level 1 – for example, *Common Nouns (CN), Proper Nouns (PN)* and *Action Verbs (AV)*; the six main question words: *How, Who, When, Where, What* and *Which* have also been added to this level; (3) Level 3 consists of all the domain-specific categories – for example, *Proper Noun Celebrity* (PN_C) and *Proper Noun Geographical Areas* (PN_G). A list of all the syntactic categories and corresponding acronyms is displayed in Appendix A.

3.2 Domain-specific syntax-based parsing and tagging

We proposed a Domain-Specific Syntax-based Parsing and Tagging (DSSPT), shown in Figure 1, for the objective of assigning not just PoS tags but also domain specific ones to help in the categorization and classification of text in different domains. The aim of this approach is to create a simple parser and tagger that could easily be applied to different domains by creating domain specific grammatical rules, in which each text is transformed to a domain-specific category using these rules. The grammatical rules contain in addition to typical categories of English grammar, domain-related grammatical categories. The domain specific syntax based parsing and tagging (DSSPT) is described below.

Phase 1: Grammar: In this phase input text is analyzed using domain knowledge and term taxonomy; this is done by identifying each keywords and phrases using the proposed tag-set. Next, the grammar is generated by identifying terminal and non-terminals nodes; the grammar in this phase is based on the Context-Free Grammar (CFG) which capture and combine two different components, i.e. the sentence structure and domain knowledge.

The target in this paper is to use a simple version of the English grammar combined with domain-specific syntactic categories since most domains do not follow entirely the formal English grammar and natural language.

Creating the grammatical rules helps with the identification of ambiguous terms since two different sentences may have similar terms but different structures, each having a different

[1] http://www.researchpipeline.com/mediawiki/index.php?title=AOL_Search_Query_Logs
[2] http://trec.nist.gov/data/million.query09.html
[3] https://ciir.cs.umass.edu/downloads/nfL6/
[4] http://trec.nist.gov/data/qa/t2007_qadata.html
[5] https://www.cs.cmu.edu/~ark/QA-data

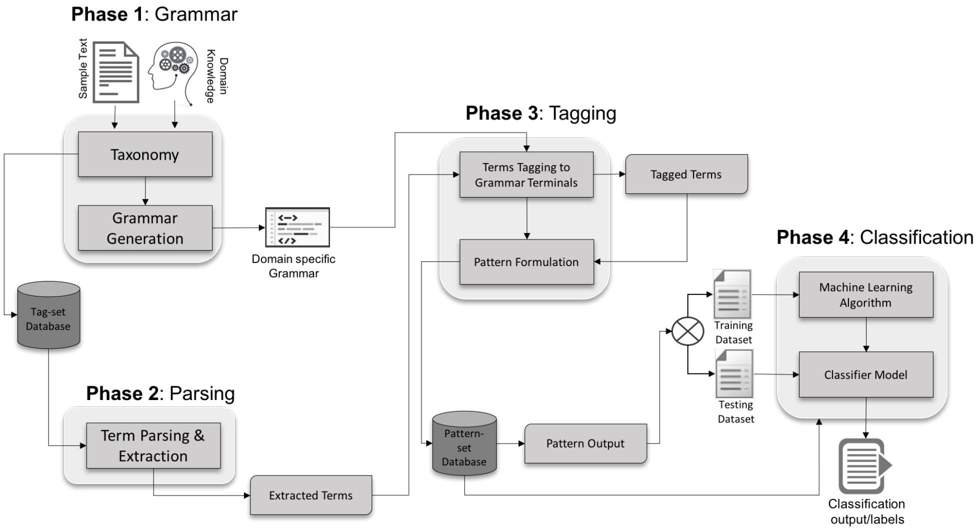

Figure 1 Framework.

meaning, which may lead to different intents. For the given examples *"Order Ed Sheeran Albums"* and *"Ed Sheeran Albums Order"*, the grammatical rules will identify the structure of the sentence at three levels: (1) at phrase level, (2) at words level which includes word classes and sub-classes and (3) domain-specific level. At phrase level, *"Order Ed Sheeran Albums"* consists of *Verb Phrase* and *Noun Phrases*, while at word level, it consists of *Verb (Action Verb)* and *Nouns (Proper Noun and Common Noun)*. At the domain specific level it consists of *Action Verb - Interact (AV_I)*, *Proper Noun - Celebrity (PN_C)* and *Common Noun - Other - Plural* (CN_{OP}). On the contrary, at phrase level, *"Ed Sheeran Albums Order"* consists of *Noun Phrases*; at word level, it consists of *Nouns (Proper Noun and Common Nouns)*. At the domain-specific level it consists of *Proper Noun - Celebrity (PN_C)*, *Common Noun - Other - Plural (CN_{OP})* and *Common Noun - Other - singular (CN_{OS})*. The different syntactical structure of the two sentence leads to different syntactical patterns, which result in different meaning, intent and search results.

Phase 2: Parsing: This step is mainly responsible for extracting terms in the text to help generate the grammar structure in the next phase to facilitate the tagging of each word to the right term category. This is done by using the keywords and phrases that have been identified from the previous phase; first, compound words will be parsed and extracted, followed by single words.

Phase 3: Tagging: In this phase the text is transformed into a pattern of grammatical terms by mapping each term to its grammar terminals; each term will be mapped to its highest level of abstraction (word class, sub-class or domain-specific) and after mapping each terms the grammatical pattern is formulated. Using the domain-specific grammar that has been generated in Phase 1 (Grammar), terms will be tagged to their terminals.

Phase 4: Classification: In this phase the patterns generated in the tagging phase are used for machine learning; the aim of this phase is to build a model for automatic classification. The classification is done by following the standard process for machine learning, which involves the splitting of the dataset into a training dataset and a test dataset. The training dataset is used for building the model, and the test dataset is used to evaluate the performance of the model.

4 Experimental Study and Results

The objective of the experimental study is to investigate the ability of our proposed parsing and tagging approach to work on different domains. Two domains were used: classification of search queries and classification of questions (for question-answering systems). To assess the performance of the machine learning classifiers, the Weka[6] software [6] was used. The experiments were set up using the typical 10-fold cross validation and the effectiveness of the classification was evaluated based on Precision, Recall and F-Measure. The results are presented in the next sub-sections for the two domains.

4.1 Queries Classification

1953 labelled queries from [14] were used, and 4,047 queries were randomly selected from AOL 2006 dataset. Queries were classified and labelled to three different categories; these categories are based on Broder's [2] classification of web queries, which are informational, navigational and transactional.

4.1.1 Results

Table 1 presents the classification performance details (Precision, Recall and F-Measure) of the Support Vector Machine (SVM) and Naive Bayes (NB) classifiers for query classification. Results show that $DSSPT_{SVM}$ identified correctly (i.e. Recall) 99.6% of the questions, while $DSSPT_{NB}$ correctly classified 95.5% of the query. $DSSPT_{SVM}$ misclassified 0.5% of transactional queries as informational, while informational and navigational queries were 100% correctly classified. Furthermore, $DSSPT_{NB}$ incorrectly classified 4.5% of the queries – 3.4% of the informational queries were classified as transactional, and 8.5% of the transactional queries were classified as informational.

Table 1 Performance of the classifiers for Query Classification.

	$DSSPT_{SVM}$			$DSSPT_{NB}$		
Accuracy	99.6%			95.5%		
Precision	0.996			0.955		
Recall	0.996			0.955		
F-score	0.996			0.955		
Class:	P	R	F	P	R	F
Info.	0.997	0.998	0.998	0.955	0.966	0.96
Nav.	1.00	1.00	1.00	0.999	1.00	1.00
Trans.	0.972	0.955	0.964	0.935	0.915	0.925

4.2 Questions Classification

We used 1,160 questions that were randomly selected from Yahoo Non-Factoid Question Dataset[7], TREC 2007 Question Answering Data[8] and a Wikipedia dataset[9]. Questions were classified and labelled to six different categories, namely: causal, choice, confirmation (Yes-No Questions), factoid (Wh-Questions), hypothetical and list. These classifications were proposed by [16].

[6] http://www.cs.waikato.ac.nz/ml/weka/
[7] https://ciir.cs.umass.edu/downloads/nfL6/
[8] http://trec.nist.gov/data/qa/t2007_qadata.html
[9] https://www.cs.cmu.edu/~ark/QA-data

4.2.1 Results

Table 2 presents the classification performance details (Precision, Recall and F-Measure) of the SVM and NB classifiers for question classification. Results show that $DSSPT_{SVM}$ identified correctly (i.e. Recall) 88.6% of the questions, while $DSSPT_{NB}$ identified correctly 83.5% of the questions.

More specifically, looking at where the errors occur, when using $DSSPT_{SVM}$, 3.2% of the causal questions were misclassified as confirmation and 32.2% were misclassified as factoid. From the choice questions, 41.7% were misclassified as confirmation and 33.3% were misclassified as factoid. Similarly, 4% of the list questions were misclassified as confirmation and 45.5% were misclassified as factoid. These results indicate that $DSSPT_{SVM}$ could not distinguish between causal, choice and list types of questions and incorrectly classified most of them as confirmation and factoid questions. Moreover, 1.6% of confirmation questions were misclassified as factoid and less than 1% were misclassified as choice or list. For the factoid questions 4.6% were misclassified as list, 1.2% were misclassified as causal, 1% were misclassified as confirmation and less than 1% were misclassified as choice. In addition, most of the hypothetical questions, i.e. 57.1%, were misclassified as factoid.

The $DSSPT_{NB}$ classifier incorrectly classified 6.5% of the causal questions as confirmation, 80.6% as factoid and 3.2% as list. Similar to $DSSPT_{SVM}$ classifier, $DSSPT_{NB}$ could not identify choice questions and misclassified 41.7% as confirmation and 58.3% as factoid. Furthermore, 0.9% of the confirmation questions were misclassified as choice, 3.4% as factoid, 2% as hypothetical and 0.9% as list. For the factoid questions, 1.3% were misclassified as causal, 0.43% as choice, 2.5% as confirmation, 0.87% as hypothetical and 2.2% as list. Moreover, 14.3% of the hypothetical questions were misclassified as causal and 57.1% as factoid. For the list type of question $DSSPT_{NB}$ incorrectly classified 7% as confirmation and 65.3% as factoid.

Table 2 Performance of the classifiers for Question Classification.

	$DSSPT_{SVM}$			$DSSPT_{NB}$		
Accuracy:	88.6%			83.5%		
Precision:	0.88			0.814		
Recall:	0.886			0.835		
F-score:	0.881			0.818		
Class:	P	R	F	P	R	F
Causal	0.714	0.645	0.678	0.231	0.097	0.136
Choice	0.429	0.25	0.316	0.00	0.00	0.00
Conf.	0.948	0.972	0.96	0.906	0.928	0.917
Factoid	0.903	0.929	0.915	0.85	0.927	0.887
Hypo.	1.00	0.429	0.6	0.133	0.286	0.182
List	0.6	0.505	0.548	0.609	0.277	0.381

Unlike the previous approaches which focus only on the type of domain, our proposed Domain-Specific Syntax-based Parsing and Tagging (DSSPT) is a general approach for incorporating domain-specific tags, which exploits the structure of the text through using domain-specific grammatical categories and rules. Moreover, the domain-specific grammar could be easily integrated in different platforms. In addition, using syntactic categories related to different domain-specific types enable the machine learning algorithms to better differentiate between different queries/question types.

5 Conclusion and Future Work

In this paper, we proposed a domain specific syntax-based Parsing and tagging (DSSPT) approach. The grammatical rules contain in addition to typical categories of English grammar, domain-related grammatical categories. The results show that our solution led to a good performance when applying it on two different domains.

The proposed framework can be applied to other domains with similar classification problems, such as Twitter, which will be investigated in future work. In addition, we aim at examining and analyzing more datasets from different domains to enrich the tag-set which will extend the ability of our framework to be used in more domains.

References

1 Thorsten Brants. TnT: a statistical part-of-speech tagger. In *Proceedings of the sixth conference on Applied natural language processing*, pages 224–231. Association for Computational Linguistics, 2000.

2 Andrei Broder. A taxonomy of web search. In *ACM Sigir forum*, volume 36(2), pages 3–10. ACM, 2002.

3 Danqi Chen and Christopher Manning. A fast and accurate dependency parser using neural networks. In *Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP)*, pages 740–750, 2014.

4 Jesús Giménez and Lluis Marquez. Fast and accurate part-of-speech tagging: The SVM approach revisited. *Recent Advances in Natural Language Processing III*, pages 153–162, 2004.

5 Kevin Gimpel, Nathan Schneider, Brendan O'Connor, Dipanjan Das, and Mills. Part-of-speech tagging for twitter: Annotation, features, and experiments. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies*, pages 42–47. Association for Computational Linguistics, 2011.

6 Mark Hall, Eibe Frank, Geoffrey Holmes, Bernhard Pfahringer, Peter Reutemann, and Ian H Witten. The WEKA data mining software: an update. *ACM SIGKDD explorations newsletter*, 11(1):10–18, 2009.

7 Atsushi Keyaki and Jun Miyazaki. Part-of-speech tagging for web search queries using a large-scale web corpus. In *Proceedings of the Symposium on Applied Computing*, pages 931–937. ACM, 2017.

8 Terry Koo, Xavier Carreras Pérez, and Michael Collins. Simple semi-supervised dependency parsing. In *46th Annual Meeting of the Association for Computational Linguistics*, pages 595–603, 2008.

9 Jayant Krishnamurthy, Pradeep Dasigi, and Matt Gardner. Neural semantic parsing with type constraints for semi-structured tables. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 1517–1527, 2017.

10 Yoong Keok Lee, Aria Haghighi, and Regina Barzilay. Simple type-level unsupervised POS tagging. In *Proceedings of the 2010 Conference on Empirical Methods in Natural Language Processing*, pages 853–861. Association for Computational Linguistics, 2010.

11 EE Luneva, PI Banokin, VS Zamyatina, and SV Ivantsov. Natural language text parsing for social network user sentiment analysis based on fuzzy sets. In *Mechanical Engineering, Automation and Control Systems (MEACS), 2015 International Conference on*, pages 1–5. IEEE, 2015.

12 Mehdi Manshadi and Xiao Li. Semantic tagging of web search queries. In *Proceedings of the Joint Conference of the 47th Annual Meeting of the ACL and the 4th International Joint*

Conference on Natural Language Processing of the AFNLP: Volume 2-Volume 2, pages 861–869. Association for Computational Linguistics, 2009.

13 David McClosky, Eugene Charniak, and Mark Johnson. Automatic domain adaptation for parsing. In *Human Language Technologies: The 2010 Annual Conference of the North American Chapter of the Association for Computational Linguistics*, pages 28–36. Association for Computational Linguistics, 2010.

14 Marcelo Mendoza and Juan Zamora. Identifying the intent of a user query using support vector machines. In *International Symposium on String Processing and Information Retrieval*, pages 131–142. Springer, 2009.

15 A. Mohasseb, M. Bader-El-Den, H. Liu, and M. Cocea. Domain specific syntax based approach for text classification in machine learning context. In *2017 International Conference on Machine Learning and Cybernetics (ICMLC)*, volume 2, pages 658–663. IEEE Systems, Man and Cybernetics, 2017.

16 Alaa Mohasseb, Mohamed Bader-El-Den, and Mihaela Cocea. Question categorization and classification using grammar based approach. *Information Processing & Management*, 2018.

17 Alaa Mohasseb, Mohamed Bader-El-Den, Andreas Kanavos, and Mihaela Cocea. Web Queries Classification Based on the Syntactical Patterns of Search Types. In *International Conference on Speech and Computer*, pages 809–819. Springer, 2017.

18 Alaa Mohasseb, Maged El-Sayed, and Khaled Mahar. Automated Identification of Web Queries using Search Type Patterns. In *WEBIST (2)*, pages 295–304, 2014.

19 Joakim Nivre, Johan Hall, and Jens Nilsson. Maltparser: A data-driven parser-generator for dependency parsing. In *Proceedings of LREC*, volume 6, pages 2216–2219, 2006.

20 Slav Petrov, Dipanjan Das, and Ryan McDonald. A universal part-of-speech tagset. *arXiv preprint arXiv:1104.2086*, 2011.

21 Yuval Pinter, Roi Reichart, and Idan Szpektor. Syntactic Parsing of Web Queries with Question Intent. In *HLT-NAACL*, pages 670–680, 2016.

22 Helmut Schmid. Probabilistic part-ofispeech tagging using decision trees. In *New methods in language processing*, page 154, 2013.

23 Noah A Smith, Michael Heilman, and Rebecca Hwa. Question generation as a competitive undergraduate course project. In *Proceedings of the NSF Workshop on the Question Generation Shared Task and Evaluation Challenge*, 2008.

24 Richard Socher, John Bauer, Christopher D Manning, et al. Parsing with compositional vector grammars. In *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, volume 1, pages 455–465, 2013.

25 Richard Socher, Christopher D Manning, and Andrew Y Ng. Learning continuous phrase representations and syntactic parsing with recursive neural networks. In *Proceedings of the NIPS-2010 Deep Learning and Unsupervised Feature Learning Workshop*, pages 1–9, 2010.

26 Pontus Stenetorp. Transition-based dependency parsing using recursive neural networks. In *NIPS Workshop on Deep Learning*. Citeseer, 2013.

27 Xiangyan Sun, Haixun Wang, Yanghua Xiao, and Zhongyuan Wang. Syntactic Parsing of Web Queries. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 1787–1796, 2016.

28 Ivan Titov and James Henderson. Porting statistical parsers with data-defined kernels. In *Proceedings of the Tenth Conference on Computational Natural Language Learning*, pages 6–13. Association for Computational Linguistics, 2006.

29 Kristina Toutanova, Dan Klein, Christopher D Manning, and Yoram Singer. Feature-rich part-of-speech tagging with a cyclic dependency network. In *Proceedings of the 2003 Conference of the North American Chapter of the Association for Computational Linguistics* *on Human Language Technology-Volume 1*, pages 173–180. Association for Computational Linguistics, 2003.

30 Gökhan Tür, Minwoo Jeong, Ye-Yi Wang, Dilek Hakkani-Tür, and Larry P. Heck. Exploiting the semantic web for unsupervised natural language semantic parsing. In *INTERSPEECH 2012, 13th Annual Conference of the International Speech Communication Association, Portland, Oregon, USA, September 9-13, 2012*, pages 338–341. ISCA, 2012.

31 Wen-tau Yih, Ming-Wei Chang, Xiaodong He, and Jianfeng Gao. Semantic Parsing via Staged Query Graph Generation: Question Answering with Knowledge Base. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics*, volume 1, pages 1321–1331, 2015.

32 Yue Zhang and Stephen Clark. A tale of two parsers: investigating and combining graph-based and transition-based dependency parsing using beam-search. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, pages 562–571. Association for Computational Linguistics, 2008.

A Appendix: Grammar terms and corresponding abbreviations

Category Name	Abbreviation	Category Name	Abbreviation
Verbs	V	Action Verbs	AV
Action Verb-Interact terms	AV_I	Action Verb-Locate	AV_L
Action Verb- Download	AV_D	Auxiliary Verb	$AuxV$
Linking Verbs	LV	Adjective Free	Adj_F
Adjective Online	Adj_O	Adjective	Adj
Adverb	Adv	Determiner	D
Conjunction	$Conj$	Preposition	P
Domain Suffix	DS	Domain Prefix	DP
Noun	N	Pronoun	$Pron$
Numeral Numbers	NN	Ordinal Numbers	NN_O
Cardinal Numbers	NN_C	Proper Nouns	PN
Celebrities Name	PN_C	Entertainment	PN_{Ent}
Newspapers, Magazines, Documents, Books	PN_{BDN}	Events	PN_E
Companies Name	PN_{CO}	Geographical Areas	PN_G
Places and Buildings	PN_{PB}	Institutions, Associations, Clubs, Parties, Foundations and Organizations	PN_{IOG}
Brand Names	PN_{BN}	Software and Applications	PN_{SA}
Products	PN_P	History and News	PN_{HN}
Religious Terms	PN_R	Holidays, Days, Months	PN_{HMD}
Health Terms	PN_{HLT}	Science Terms	PN_S
Common Noun	CN	Common Noun – Other- Singular	CN_{OS}
Common Noun- Other- Plural	CN_{OP}	Database and Servers	CN_{DBS}
Advice	CN_A	Download	CN_D
Entertainment	CN_{Ent}	File Type	CN_{File}
Informational Terms	CN_{IFT}	Obtain Offline	CN_{OF}
Obtain Online	CN_{OO}	History and News	CN_{HN}
Interact terms	CN_I	Locate	CN_L
Site, Website, URL	CN_{SWU}	Question Words	QW
How	QW_{How}	What	QW_{What}
When	QW_{When}	Where	QW_{Where}
Who	QW_{Who}	Which	QW_{Which}