Improving Syntactic Coordination Resolution Using Language Modeling

Philip V. Ogren
Center for Computational Pharmacology
University of Colorado Denver
12801 E. 17th Ave
Aurora, CO 80045, USA
philip@ogren.info

Abstract

Determining the correct structure of coordinating conjunctions and the syntactic constituents that they coordinate is a difficult task. This subtask of syntactic parsing is explored here for biomedical scientific literature. In particular, the intuition that sentences containing coordinating conjunctions can often be rephrased as two or more smaller sentences derived from the coordination structure is exploited. Generating candidate sentences corresponding to different possible coordination structures and comparing them with a language model is employed to help determine which coordination structure is best. This strategy is used to augment a simple baseline system for coordination resolution which outperforms both the baseline system and a constituent parser on the same task.

1 Introduction

For this work, coordination resolution (CR) refers to the task of automatically identifying the correct coordination structure of coordinating conjunctions. In this study the conjunctions *and* and *or* and the conjuncts they coordinate are examined. CR is an important subtask of syntactic parsing in the biomedical domain because many information extraction tasks require correct syntactic structures to perform well, in particular coordination structures. For example, (Cohen et al., 2009) showed that using a constituent parser trained on biomedical data to provide coordination structures to a high-precision protein-protein interaction recognition system resulted in a significant performance boost from an overall F-measure of 24.7 to 27.6. Coordination structures are the source of a disproportionate number of parsing errors for both constituent parsers (Clegg and Shepherd, 2007) and dependency parsers (Nivre and McDonald, 2008).

CR is difficult for a variety of reasons related to the linguistic complexity of the phenomenon. There are a number of measurable characteristics of coordination structures that support this claim including the following: constituent types of conjuncts, number of words per conjunct, number of conjuncts per conjunction, and the number of conjunctions that are nested inside the conjunct of another conjunction, among others. Each of these metrics reveal wide variability of coordination structures. For example, roughly half of all conjuncts consist of one or two words while the other half consist of three or more words including 15% of all conjuncts that have ten or more words. There is also an increased prevalence of coordinating conjunctions in biomedical literature when compared with newswire text. Table 1 lists three corpora in the biomedical domain that are annotated with deep syntactic structures; CRAFT (described below), GENIA (Tateisi et al., 2005), and Penn BIOIE (Bies et al., 2005). The number of coordinating conjunctions they contain as a percentage of the number of total tokens in each corpus are compared with the Penn Treebank corpus (Marcus et al., 1994). The salient result from this table is that there are 50% more conjunctions in biomedical scientific text than in newswire text. It is also interesting to note that 15.4% of conjunctions in the biomedical corpora are nested inside a conjunct of another con-

Proceedings of the NAACL HLT 2010 Student Research Workshop, pages 1–6,
Los Angeles, California, June 2010.

junction as compared with 10.9% for newswire.

Table 1: Biomedical corpora that provide coordination structures compared with the Penn Treebank corpus.

Corpus	Tokens	Conjunctions	
CRAFT	246,008	7,115	2.89%
GENIA	490,970	14,854	3.03%
BIOIE	188,341	5,036	2.67%
subtotal	925,319	27,005	**2.92%**
PTB	1,173,766	22,888	**1.95%**

The Colorado Richly Annotated Full-Text (CRAFT) Corpus being developed at the University of Colorado Denver was used for this work. Currently, the corpus consists of 97 full-text open-access scientific articles that have been annotated by the Mouse Genome Institute[1] with concepts from the Gene Ontology[2] and Mammalian Phenotype Ontology[3]. Thirty-six of the articles have been annotated with deep syntactic structures similar to that of the Penn Treebank corpus described in (Marcus et al., 1994). As this is a work in progress, eight of the articles have been set aside for a final holdout evaluation and results for these articles are not reported here. In addition to the standard treebank annotation, the *NML* tag discussed in (Bies et al., 2005) and (Vadas and Curran, 2007) which marks nominal subconstituents which do not observe the right-branching structure common to many (but not all) noun phrases is annotated. This is of particular importance for coordinated noun phrases because it provides an unambiguous representation of the correct coordination structure. The coordination instances in the CRAFT data were converted to simplified coordination structures consisting of conjunctions and their conjuncts using a script that cleanly translates the vast majority of coordination structures.

2 Related Work

There are two main approaches to CR. The first approach considers CR as a task in its own right where the solutions are built specifically to perform CR. Often the task is narrowly defined, e.g. only coordinations of the pattern *noun-1 conjunction noun-2 noun-3* are considered, and relies on small training and testing data sets. Generally, such research efforts do not attempt to compare their results with previous results other than in the broadest and most-qualified way. Studies by (Chantree et al., 2005), (Nakov and Hearst, 2005), and (Resnik, 1999) are representative examples of such work. A study by (Shimbo and Hara, 2007) performed CR on sentences from the GENIA corpus containing one instance of the word "and" coordinating noun phrases. They used a sequence alignment algorithm modified for CR drawing on the intuition that conjuncts have similar syntactic constructs. In each of these studies, promising results were achieved by careful application of their respective approaches. However, each study is limited in important respects because they narrowly constrain the problem, use limited training data, and make certain unrealistic assumptions in their experimental setup that make general application of their solutions problematic. For example, in the study by (Shimbo and Hara, 2007) they chose only sentences that have one instance of "and" because their algorithm does not handle nested conjunctions. Additionally, they assume an oracle that provides the system with only sentences that contain coordinated noun phrases.

The work most similar to this study was done by (Hara et al., 2009) in that they define the CR task essentially the same as is done here. Their approach involves a grammar tailored for coordination structures that is coupled with a sequence alignment algorithm that uses perceptrons for learning feature weights of an edit graph. The evaluation metric they use is slightly less strict than the metric used for this study in that they require identification of the left boundary of the left-most conjunct and the right boundary of the right-most conjunct to be counted correct. Two other important differences are that the evaluation data comes from the GENIA corpus and they use gold-standard part-of-speech tags for the input data. Regardless of these relatively minor differences, their performance of 61.5 F-measure far outperforms what is reported below and experiments that are directly comparable to their work will be performed.

[1] http://www.informatics.jax.org/

[2] http://geneontology.org/

[3] http://www.informatics.jax.org/searches/MP_form.shtml

The second main approach considers CR within the broader task of syntactic parsing. Any syntactic parser that generates constituents or dependencies must necessarily perform CR to perform well. Typically, a syntactic parser will have a single, central algorithm that is used to determine all constituents or dependencies. However, this does not preclude parsers from giving special attention to CR by adding CR-specific rules and features. For example, (Nilsson et al., 2006) show that for dependency parsing it is useful to transform dependency structures that make conjunctions the head of their conjuncts into structures in which coordination dependencies are chained. (Charniak and Johnson, 2005) discusses a constituent-based parser that adds two features to the learning model that directly address coordination. The first measures parallelism in the labels of the conjunct constituents and their children and the second measures the lengths of the conjunct constituents. The work done by (Hogan, 2007) focuses directly on coordination of noun phrases in the context of the Collins parser (Collins, 2003) by building a right conjunct using features from the already built left conjunct.

3 Using a Language Model

Consider the following sentence:

> Tyr mutation results in *increased IOP* and *altered diurnal changes*.

By exploiting the coordination structure we can rephrase this sentence as two separate sentences:

- Tyr mutation results in *increased IOP*.
- Tyr mutation results in *altered diurnal changes*.

Using this simple rewrite strategy a candidate sentence for each possible conjunct can be composed. For this sentence there are six possible left conjuncts corresponding to each word to the left of the conjunction. For example, the candidate conjunct corresponding to the third word is *results in increased IOP* and the corresponding sentence rewrite is *Tyr mutation altered diurnal changes.* The resulting candidate sentences can be compared by calculating a sentence probability using a language model. Ideally, the candidate sentence corresponding to the correct conjunct boundary will have a higher probability than the other candidate sentences. One problem with this approach is that the candidate sentences are different lengths. This has a large and undesirable (for this task) impact on the probability calculation. A simple and effective way to normalize for sentence length is by adding[4] the probability of the candidate conjunct (also computed by using the language model) to the probability of the candidate sentence. The probability of each candidate is calculated using this simple metric and then rank ordered. Because the number of candidate conjuncts varies from one sentence to the next (as determined by the token index of the conjunction) it is useful to translate the rank into a percentile. The rank percentile of the candidate conjuncts will be applied to the task of CR as described below. However, it is informative to directly evaluate how good the rank percentile scores of the correct conjuncts are.

To build a language model a corpus of more than 80,000 full-text open-access scientific articles were obtained from PubMed Central[5]. The articles are provided in a simple XML format which was parsed to produce plain text documents using only sections of the articles containing contentful prose (i.e. by excluding sections such as e.g. *acknowledgments* and *references*.) The plain text documents were automatically sentence segmented, tokenized, and part-of-speech tagged resulting in nearly 13 million sentences and over 250 million tagged words. A language model was then built using this data with the SRILM toolkit described in (Stolcke, 2002). Default options were used for creating the language model except that the order of the model was set to four and the "-tagged" option was used. Thus, a 4-gram model with Good-Turing discounting and Katz backoff for smoothing was built.

For each token to the left of a conjunction a candidate conjunct/sentence pair is derived, its probability calculated, and a rank percentile score is assigned to it relative to the other candidates. Because multiple conjuncts can appear on the left-hand-side of the conjunction, the left border of the leftmost conjunct is considered here. The same is done for tokens

[4] logprobs are used here

[5] `http://www.ncbi.nlm.nih.gov/pmc/about/ftp.html`. The corpus was downloaded in September of 2008.

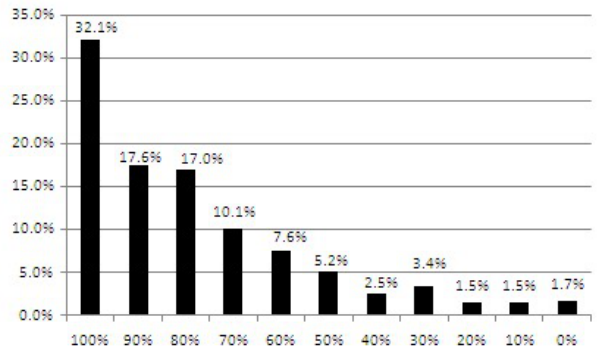

Figure 1: The first column can be read as "The correct conjunct candidate had the highest rank percentile 32.1% of the time." The second column can be read as "The correct conjunct candidate had a rank percentile of 90% or greater 17.6% of the time." The columns add to one.

on the right-hand-side of the conjunction. Figure 1 shows a histogram of the rank percentile scores for the correct left conjunct. The height of the bars correspond to the percentage of the total number of conjunctions in which the correct candidate was ranked within the percentile range. Thus, the columns add to one and generalizations can be made by adding the columns together. For example, 66.7% of the conjunctions (by adding the first three columns) fall above the eightieth percentile. The overall average rank percentage for all of the left-hand-side conjuncts was 81.1%. The median number of candidates on the left-hand-side is 17 (i.e. the median token index of the conjunction is 17). Similar results were obtained for the right-hand-side data but were withheld for space considerations. The overall average rank percentage for right-hand-side conjuncts was 82.2%. This slightly better result is likely due to the smaller median number of candidates on the right-hand-side of 12 (i.e. the median token index of the conjunction is 12 from the end of the sentence.) These data suggest that the rank percentile of the candidate conjuncts calculated in this way could be an effective feature to use for CR.

4 Coordination Resolution

Table 2 reports the performance of two CR systems that are described below. Results are reported as F-Measure at both the conjunct and conjunction levels where a true positive requires all boundaries to be exact. That is, for conjunct level evaluation a conjunct generated by the system must have exactly the same extent (i.e. character offsets) as the conjunct in the gold-standard data in addition to being attached to the same conjunction. Similarly, at the conjunction level a true positive requires that a coordination structure generated by the system has the same number of conjuncts each with extents exactly the same as the corresponding conjunct in the gold-standard coordination structure. Where 10-fold cross-validation is performed, training is performed on roughly 90% of the data and testing on the remaining 10% with the results micro-averaged. Here, the folds are split at the document level to avoid the unfair advantage of training and testing on different sections of the same document.

Table 2: Coordination resolution results at the conjunct and conjunction levels as F-Measure.

	Conjunct	Conjunction
OpenNLP + PTB	55.46	36.56
OpenNLP + CRAFT	58.87	39.50
baseline	59.75	40.99
baseline + LM	**64.64**	**46.40**

The first system performs CR within the broader task of syntactic parsing. Here the constituent parser from the OpenNLP project[6] is applied. This parser was chosen because of its availability and ease of use for both training and execution. It has also been shown by (Buyko et al., 2006) to perform well on biomedical data. The output of the parser is processed by the same conversion script described above. The parser was trained and evaluated on both the Penn Treebank and CRAFT corpora. For the latter, 10-fold cross-validation was performed. Preliminary experiments that attempted to add additional training data from the GENIA and Penn BIOIE corpora proved to be slightly detrimental to performance in both cases. Table 2 shows that CR improves at the conjunction level by nearly three points (from 36.56 to 39.50) by simply training on biomedical data rather than using a model trained on newswire.

The second system that performs CR as a separate

[6] `http://opennlp.sf.net`

task by using token-level classification to determine conjunct boundaries is introduced and evaluated. In brief, each token to the left of a conjunction is classified as being either a left-hand border of a conjunct for that conjunction or not. Similarly, tokens to the right of a conjunction are classified as either a right-hand border of a conjunct or not. From these token-level classifications and some simple assumptions about the right-hand and left-hand borders of left and right conjuncts, respectively,[7] a complete coordination structure can be constructed. The classifier used was SVM^{light} described in (Joachims, 1999) using a linear kernel. The baseline system uses a number of shallow lexical features (many common to named entity recognition systems) including part-of-speech tags, word and character n-grams, the distance between the focus token and the conjunction, and word-level features such as whether the token is a number or contains a hyphen. A more detailed description of the baseline system is avoided here as this remains a major focus of current and future research efforts and the final system will likely change considerably. Table 2 shows the results of 10-fold cross-validation for the baseline system. This simple baseline system performs at 40.99 F-measure at the conjunction level which is modestly better than the syntactic parser trained on CRAFT.

The baseline system as described above was augmented using the language modeling approach described in Section 3 by adding a simple feature to each token being classified whose value is the rank percentile of the probability of the corresponding conjunct candidate. Again, 10-fold cross-validation was performed. Table 2 shows that this augmented baseline system performs at 46.40 F-measure at the conjunction level which out-performs the baseline system and the CRAFT-trained parser by 5.4 and 6.9 points, respectively. This increase in performance demonstrates that a language model can be effectively purposed for CR.

While the use of a language model to improve CR results is promising, the results in Table 2 also speak to how difficult this task is for machines to perform. In contrast, the task is comparatively easy for humans to perform consistently. To calculate inter-annotator agreement on the CR task, 500 sentences containing either the word "and" or "or" were randomly chosen from the 13 million sentence corpus described in Section 3 and annotated with coordination structures by two individuals, the author and another computer scientist with background in biology. Our positive specific agreement[8] was 91.93 and 83.88 at the conjunct and conjunction level, respectively, for 732 conjunctions. This represents a dramatic gulf between system and human performance on this task but also suggests that large improvements for automated CR should be expected.

5 Future Work

There is much that can be done to move this work forward. Creating comparable results to the study discussed in Section 2 by (Hara et al., 2009) is a top priority. As alluded to earlier, there is much that can be done to improve the baseline system. For example, constraining coordination structures to not overlap except where one is completely nested within the conjunct of another should be enforced as partially overlapping coordination structures never occur in the training data. Similarly, a conjunction that appears inside parentheses should have a coordination structure that is completely contained inside the parentheses. Thorough error analysis should also be performed. For example, it would be interesting to characterize the conjuncts that have a low rank percentile for their calculated probability. Also, it would be useful to measure performance across a number of metrics such as phrase type of the conjuncts, length of conjuncts, whether a coordination structure is nested inside another, etc. Demonstrating that CR can improve syntactic parsing performance and improve the performance of an information extraction system would give this work greater significance.

Conclusion

This work has demonstrated that a language model can be used to improve performance of a simple CR system. This is due to the high rank percentile of the probability of the correct conjunct compared with other possible conjuncts.

[7] For example, the left-hand border of the conjunct to the right of a conjunction will always be the first word following the conjunction.

[8] This measure is directly comparable with F-measure.

References

Ann Bies, Seth Kulick, and Mark Mandel. 2005. Parallel entity and treebank annotation. In *CorpusAnno '05: Proceedings of the Workshop on Frontiers in Corpus Annotations II*, pages 21–28, Morristown, NJ, USA. Association for Computational Linguistics.

Ekaterina Buyko, Joachim Wermter, Michael Poprat, and Udo Hahn. 2006. Automatically adapting an NLP core engine to the biology domain. In *Proceedings of the Joint BioLINK-Bio-Ontologies Meeting. A Joint Meeting of the ISMB Special Interest Group on Bio-Ontologies and the BioLINK Special Interest Group on Text Data M ining in Association with ISMB*, pages 65–68. Citeseer.

Francis Chantree, Adam Kilgarriff, Anne De Roeck, and Alistair Willis. 2005. Disambiguating coordinations using word distribution information. *Proceedings of RANLP2005*.

Eugene Charniak and Mark Johnson. 2005. Coarse-to-fine n-best parsing and maxent discriminative reranking. In *Proceedings of the 43rd Annual Meeting of the Association for Computational Linguistics (ACL'05)*, pages 173–180, Ann Arbor, Michigan, June. Association for Computational Linguistics.

Andrew Clegg and Adrian Shepherd. 2007. Benchmarking natural-language parsers for biological applications using dependency graphs. *BMC Bioinformatics*, 8(1):24.

Kevin B. Cohen, Karin Verspoor, Helen L. Johnson, Chris Roeder, Philip V. Ogren, William A. Baumgartner Jr, Elizabeth White, Hannah Tipney, and Lawrence Hunter. 2009. High-precision biological event extraction with a concept recognizer. In *Proceedings of the Workshop on BioNLP: Shared Task*, pages 50–58. Association for Computational Linguistics.

Michael Collins. 2003. Head-driven statistical models for natural language parsing. *Computational linguistics*, 29(4):589–637.

Kazuo Hara, Masashi Shimbo, Hideharu Okuma, and Yuji Matsumoto. 2009. Coordinate structure analysis with global structural constraints and alignment-based local features. In *ACL-IJCNLP '09: 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*, pages 967–975, Morristown, NJ, USA. Association for Computational Linguistics.

Deirdre Hogan. 2007. Coordinate noun phrase disambiguation in a generative parsing model. In *Proceedings of the 45th Annual Meeting of the Association of Computational Linguistics*, pages 680–687, Prague, Czech Republic, June. Association for Computational Linguistics.

Thorsten Joachims. 1999. Making large scale SVM learning practical.

Mitchell P. Marcus, Beatrice Santorini, and Mary Ann Marcinkiewicz. 1994. Building a large annotated corpus of English: The Penn Treebank. *Computational linguistics*, 19(2):313–330.

Preslav Nakov and Marti Hearst. 2005. Using the web as an implicit training set: Application to structural ambiguity resolution. In *Proceedings of Human Language Technology Conference and Conference on Empirical Methods in Natural Language Processing*, pages 835–842, Vancouver, British Columbia, Canada, October. Association for Computational Linguistics.

Jens Nilsson, Joakim Nivre, and Johan Hall. 2006. Graph transformations in data-driven dependency parsing. In *Proceedings of the 21st International Conference on Computational Linguistics and 44th Annual Meeting of the Association for Computational Linguistics*, pages 257–264, Sydney, Australia, July. Association for Computational Linguistics.

Joakim Nivre and Ryan McDonald. 2008. Integrating graph-based and transition-based dependency parsers. In *Proceedings of ACL-08: HLT*, pages 950–958, Columbus, Ohio, June. Association for Computational Linguistics.

Philip Resnik. 1999. Semantic similarity in a Taxonomy: An Information-Based Measure and its Application to Problems of Ambiguity in Natural Language. *Journal of Artificial Intelligence*, 11(11):95–130.

Masashi Shimbo and Kazuo Hara. 2007. A discriminative learning model for coordinate conjunctions. In *Proceedings of the 2007 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning (EMNLP-CoNLL)*, pages 610–619.

Andreas Stolcke. 2002. SRILM-an extensible language modeling toolkit. In *Seventh International Conference on Spoken Language Processing*, volume 3. Citeseer.

Yuka Tateisi, Akane Yakushiji, Tomoko Ohta, and Junichi Tsujii. 2005. Syntax Annotation for the GENIA corpus. In *Second International Joint Conference on Natural Language Processing (IJCNLP05)*, pages 222–227.

David Vadas and James Curran. 2007. Adding noun phrase structure to the penn treebank. In *Proceedings of the 45th Annual Meeting of the Association of Computational Linguistics*, pages 240–247, Prague, Czech Republic, June. Association for Computational Linguistics.