

Automatic Measurement of Syntactic Complexity Using the Revised Developmental Level Scale

Xiaofei Lu

Department of Applied Linguistics
The Pennsylvania State University
University Park, PA 16802, USA
xxl13@psu.edu

Abstract

This paper describes a heuristics-based system for automatic measurement of syntactic complexity using the revised Developmental Level (D-Level) Scale (Rosenberg and Abbeduto, 1987; Covington et al., 2006). The system takes a raw sentence as input and assigns it to an appropriate developmental level (0-7). The system is designed with child language acquisition and psycholinguistic research in mind, and is therefore developed and evaluated using child language acquisition data from the CHILDES database (MacWhinney, 2000). Experiment results show that the model achieves an accuracy of 93.2% on unseen spoken data.

Introduction

Syntactic complexity refers to “the range of forms that surface in language production and the degree of sophistication of such forms” (Ortega, 2003, p. 492). Syntactic complexity metrics have proven to be important research tools in a variety of language-related research areas, such as child language acquisition, language impairment, language and aging, and second language acquisition. A number of syntactic complexity metrics have been proposed for psycholinguistic and/or child language acquisition research, e.g., mean length of utterance (MLU), Developmental Sentence Scoring (DSS) (Lee, 1974), Index of Productive Syntax (IPSyn) (Scarborough, 1990), and Developmental Level (D-Level) (Rosenberg and Abbeduto, 1987), among others. Manual analysis of syntactic complexity of large language samples is an extremely laborious process, requiring skilled analysts to identify a range of relevant syntactic constructions in the sample. This has limited the size of the language samples analyzed in previous research. There is a

clear need for computational tools that can automate the process with high accuracy.

There have been a few recent endeavors to automate the process. These endeavors differ with respect to the kinds of complexity metrics implemented as well as the levels of natural language processing (NLP) capabilities incorporated. Long, Fey and Channell (2004) implemented a software package, computerized profiling, for child language study, which includes automated computation of IPSyn and DSS using part-of-speech (POS) and morphological analysis. Sagae et al. (2005) described a system that automatically analyzes the complexity of sentences using IPSyn. This system makes use of a statistical parser and achieves higher accuracy than computerized profiling. Graesser et al. (2004) developed a software package, Coh-Metrix, which calculates the coherence of texts using a wide range of measures, including three indexes of syntactic complexity: mean number of modifiers per noun phrase; mean number of higher level constituents per sentence, controlling for number of words; and the number of words that appear before the main verb of the main clause in the sentences.

This paper describes a heuristics-based system for automatic measurement of syntactic complexity using the revised D-Level Scale (Covington et al., 2006). The system takes a raw sentence as input and assigns it to an appropriate developmental level (0-7). The system is designed with psycholinguistic and child language acquisition research in mind, and is therefore developed and evaluated using spoken child-adult and child-child interaction data from the CHILDES database (MacWhinney, 2000).

The rest of the paper is organized as follows. Section 2 describes the D-Level Scale and the motivation for implementing this metric. Section 3 details the specifics of the heuristics-based system for automatic syntactic complexity analysis. Section 4 reports results of the model on child language acquisition data from the CHILDES database as well as findings from the error analysis. Section 5 concludes the paper with a discussion of the implications of the results and avenues for further research.

The Developmental Level Scale

The original D-Level Scale was proposed by Rosenberg and Abbeduto (1987) based primarily on observations of child language acquisition. The scale classifies certain types of complex sentences into seven developmental levels, but many types of sentences are left out. Covington et al. (2006) extended the scale to cover all sentences along with minor revisions of the arrangement of certain levels based on recent psycholinguistic research. The revised D-Level Scale consists of the following eight levels: (0) simple sentences, including questions; sentences with auxiliaries and semi-auxiliaries; simple elliptical sentences; (1) infinitive or *-ing* complement with same subject as main clause; (2) conjoined noun phrases in subject position; sentences conjoined with a coordinating conjunction; conjoined verbal, adjectival, or adverbial construction; (3) relative or appositional clause modifying object of main verb; nominalization in object position; finite clause as object of main verb; subject extraposition; (4) non-finite complement with its own understood subject; comparative with object of comparison; (5) sentences joined by a subordinating conjunction; nonfinite clauses in adjunct positions; (6) relative or appositional clause modifying subject of main verb; embedded clause serving as subject of main verb; nominalization serving as subject of main verb; (7) more than one level of embedding in a single sentence. Elliptical sentences are rated according to what is actually uttered, not what could have been uttered.

The motivations for implementing a system for automatic syntactic complexity measurement using the D-Level Scale are threefold. First, given that the system is developed with child language acquisition research in mind, the D-Level Scale is especially pertinent as it is “the only acquisition-based sentence complexity scale in current use” (Covington et al., 2006, p. 1). Second, the D-Level Scale has also been shown by psycholinguistic experiments to be a more adequate index of sentence comprehension and recall than many other metrics, such as MLU and Proposition Density (Cheung and Kemper, 1992). Third, a computational system for D-Level analysis, along with existing ones for other metrics discussed above, will facilitate a large-scale, empirical evaluation of the validity of the wide range of metrics in current use. This evaluation is critical given that syntactic complexity measures have been used to study important issues such as relations between syntactic acquisition in early life and symptoms of Alzheimer’s disease in old age (Snowdon et al., 1996).

The System

To rate the complexity of an input sentence using the D-Level Scale, it is essential to identify each of the syntactic constructions defined in the scale in the sentence. The system achieves this in two stages. In the preprocessing stage, it utilizes existing state-of-the-art NLP technology to

first assign each token in the input sentence a tag that indicates its POS category and then parse the sentence. The output is an analysis of the syntactic structure of the sentence in the form of a parse tree. In the syntactic complexity analysis stage, the system analyzes the parse tree and assigns the sentence to an appropriate developmental level, in three steps. First, the parse tree is decomposed into its component subtrees. Second, each subtree is scored based on the type or types of syntactic construction detected in it using a set of heuristics. Finally, the sentence is assigned to an appropriate developmental level based on the scores assigned to the subtrees.

Preprocessing

As the syntactic complexity analyzer processes a language sample sentence by sentence, the sample needs to be segmented into individual sentences first. A number of sentence segmentation systems exist. In the CHILDES databases, however, this is already done manually by the transcribers. We use Tsuruoka and Tsujii’s (2005) POS tagger to assign each token in the input sentence a tag that indicates its POS category. This tagger has a built-in tokenizer that segments the sentence into individual tokens. It achieves a tagging accuracy of 97.15% on Sections 22-24 of the Wall Street Journal (WSJ) portion of the Penn Treebank (Marcus et al., 1993). While it is possible to recognize some syntactic constructions using POS information alone, many complex structures require deeper syntactic analysis than POS information. We use Collins’ (1999) statistical parser to analyze the syntactic structure of sentences. The parser achieves over 88% precision and recall for constituent labeling on the WSJ corpus.

Syntactic Complexity Analysis

Parse Tree Decomposition. Given the parse tree generated by the preprocessor, the syntactic complexity analyzer first traverses it to decompose it into its component subtrees. Each subtree consists of a root node and an ordered list of its immediate children. The root node is uniquely identified with the following six pieces of information: its phrasal category, its lexical head, the number of children it has, the position of its head child in the list of children, its depth in the parse tree, and its order among all the subtrees at the same depth level. This information allows the system to uniquely identify each subtree as well as to track the dominance relationship among the subtrees.

Several constructions defined in the D-Level Scale involve identifying noun phrases in subject or object position. The system retrieves this information at the same time it extracts the subtrees. In general, a noun phrase is in subject position if it is pre-verbal and is immediately dominated by a clausal node, and it is in object position if it is immediately dominated by a VP node.

Heuristics-Based Subtree Processing. In the second step, the system processes the subtrees one by one using a set of heuristics. Each heuristic defines a pattern for a particular syntactic construction and assigns a subtree in which the construction is found to the corresponding developmental level. In many cases, it is possible to identify the structure of a subtree by analyzing the subtree itself. However, it is sometimes necessary to examine subtrees at lower levels in order to determine the structure of the current subtree.

The heuristics are organized around subtrees with clausal, NP, VP, ADJP and ADVP root nodes. A subtree with only one leaf is assigned to level 0 without further checking. For a subtree with more than one leaves, the category of its root node is determined, and the appropriate set of heuristics is then applied. If no construction from a non-zero level is found, the subtree is assigned to level 0. If one or more constructions from the same non-zero level are found, the subtree is assigned to that non-zero level. If two or more constructions from two different non-zero levels are found, the subtree is assigned to level 7.

The following heuristics are applied to subtrees with a clausal root node, including S (simple declarative clause), SBAR (clause introduced by a subordinating conjunction), SBARQ (direct question introduced by a *wh*-word or a *wh*-phrase), SG (non-finite clause), SINV (inverted declarative sentence), and SQ (inverted yes/no question or main clause of a *wh*-question).

- (1) There is an embedded clause in subject position (level 6) if the root is not a top-level SG node, there is a VP child but no NP child, and there is either exactly one pre-verbal clausal child or a pre-verbal clausal argument child¹.
- (2) There are sentences joined by a subordinating conjunction (level 5) if there is a non-argument SBAR child, a VP child, and either a pre-verbal NP child or a non-argument pre-verbal clausal child.
- (3) There is a non-finite clause in adjunct position (level 5) if
 - a. There is a non-argument SG child, a VP child, and either a pre-verbal NP child or a pre-verbal clausal argument child; or
 - b. There is a PP child that has an SG-A child; or
 - c. There is a PP child whose headword is the present or past participle form of a verb.
- (4) There is a finite clause as object of main verb (level 3) if the root node is of category SINV and there is a pre-verbal S-A child.
- (5) There are sentences conjoined with coordinating conjunction (CC) (level 2) if there is a CC child between two clausal children.
- (6) There is a conjoined adverbial construction (level 2) if the root node is of category SBARQ and there is a CC child between two WHADVP children.

¹ Arguments are marked with “-A” in the parse tree.

Some heuristics for processing subtrees with an NP node as mother entail a method for nominalization recognition. Nominalizations are sentences encapsulated into an abstract noun phrase. Covington et al. (2006) considered an NP as a nominalization only if it includes explicitly the subject and/or object of the root verb. Thus, *the construction* is not a nominalization, but *the city's construction of the roads* is. To capture this definition, the system accesses NOMLEX (Macleod et al., 1998), a lexicon in which each entry contains comprehensive information about a nominalization, including its root verb, a list of nominal positions where the verbal subject can be found, and the types and positions of allowable nominalization complements. This information allows the system to determine whether a subject or object NP is a nominalization according to the definition in the D-Level Scale by first looking up the lexical head of the NP in the lexicon and then checking whether the subject and/or object of the associated verb is realized in the NP.

The following heuristics are applied to subtrees with a NP root node.

- (7) There is a relative clause modifying subject of main verb (level 6) if the NP root node is subject of main verb and there is either a clausal child or a VP child.
- (8) There is an appositional clause modifying subject of main verb (level 6) if the NP root node is subject of main verb and the child node from which the head noun is taken is followed by an NP child node that has either a clausal child or a VP child.
- (9) There is a nominalization serving as subject of main verb (level 6) if the NP root node is subject of main verb and is a nominalization.
- (10) There is a relative clause modifying object of main verb (level 3) if the NP root node is object of main verb and there is either a clausal child or a VP child.
- (11) There is an appositional clause modifying object of main verb (level 3) if the NP root node is object of main verb and the child node from which the head noun is taken is followed by an NP child node that has either a clausal child or a VP child.
- (12) There is a nominalization in object position (level 3) if the NP root node is in object position and is a nominalization.
- (13) There is a conjoined noun phrase in subject position (level 2) if the NP root node is in subject position and there is a CC child between two noun or NP children.
- (14) There is a conjoined adjectival construction (level 2) if there is a CC child between two adjective children.

The following heuristics are applied to subtrees with a VP root node.

- (15) There are sentences conjoined by a subordinating conjunction (level 5) if there is a non-argument SBAR child or an ADVP child that has an SBAR child.

- (16) There is a non-finite clause in adjunct position (level 5) if
- There is a non-argument SG child; or
 - There is a PP child that has an SG-A child; or
 - There is a PP child whose headword is the present or past participle form of a verb.
- (17) There is a non-finite complement with its own subject (level 4) if
- There is a post-verbal clausal argument child that is not headed by a finite verb and that has a pre-verbal subject child; or
 - There is a post-verbal NP child followed by an SG-A child headed by *to*; or
 - There is post-verbal NP child followed by an ADJP child; or
 - There are two consecutive post-verbal NP children and the second is not a temporal NP; or
 - There is a post-verbal NP child followed by a PP child whose headword is *into*; or
 - The head verb of the VP is a verb that may require both an NP and a PP complement² and there is a post-verbal NP child followed by a PP child that does not have any clausal child.
- (18) There is a raising construction (level 3) if there is a post-verbal PP child whose headword is *to* that is immediately followed by an SBAR-A child whose headword is also *to*.
- (19) There is a finite clause as object of main verb (level 3) if there is a post-verbal SBAR-A child or a clausal argument child whose headword is a finite verb.
- (20) There is a conjoined verbal construction (level 2) if there is a CC child between two verb or VP children.
- (21) There is an infinitive complement with same subject as main verb (level 1) if the VP root node is not headed by an auxiliary verb or *going*, and there is a post-verbal clausal argument child headed by *to*.
- (22) There is an *-ing* complement with same subject as main verb (level 1) if the VP root node is not headed by an auxiliary verb and there is a VP or clausal argument child that has an *-ing* form headword and that has no pre-verbal subject child.

Finally, the following three heuristics are applied to subtrees with an ADJP or ADVP root node.

- (23) There is a comparative with object of comparison (level 4) if
- There is a PP or SBAR child headed by *than*; or
 - There is an adjective or adverb child that is preceded by *as* or *so* and followed by a PP child headed by *as*; or
 - There is an ADVP child headed by *as* that is followed by a PP child headed by *as*.

² The list currently contains the following verbs: *deem*, *leave*, *mark*, *name*, *put*, and *set*.

- (24) There is subject extraposition (level 3) if the root node is of category ADJP and there is an SBAR child that has an S-A child whose headword is *to*.
- (25) There is a conjoined adjectival construction (level 2) if the root node is of category ADJP and there is a CC child between two adjective or ADJP children.
- (26) There is a conjoined adverbial construction (level 2) if the root node is of category ADVP and there is a CC child between two adverbial or ADVP children.

Sentence Complexity Scoring. After all subtrees in a parse tree are scored using the heuristics, the syntactic complexity analyzer assigns the sentence to an appropriate developmental level as follows. If all subtrees are assigned to level zero, the sentence is assigned to level 0; if one and only one non-zero level is assigned to one or more subtrees, the sentence is assigned to that non-zero level; if two or more different non-zero scores are assigned to two or more of the subtrees, the sentence is assigned to level 7.

Results

Experiment setup

The system is developed and evaluated using data from the CHILDES database, which consists of transcriptions of spoken interactions between adults and young children of different ages. To ensure that sentences of different levels occur in the data, we randomly selected a file (boys90.cha in the MacWhinney folder) in which the target children are of ages 7 and 5 respectively, as we suspected younger children may produce fewer sentences at higher levels. All lines that contain material other than actual transcriptions were removed. Characters that mark emphasis, repetition, etc. in the utterances were also removed. In addition, utterances that contain less than three words were discarded. The training and test dataset each consists of 500 sentences randomly selected from the formatted file. The training data is used for developing the heuristics, while the test data is used to evaluate the performance of the system on unseen data. The sentences were independently rated by two annotators. Inter-annotator agreement using Cohen's kappa was very high (kappa = 0.9108), consistent with the high inter-annotator reliability reported in Cheung and Kemper (1992) and Covington et al. (2006). The annotators reconciled differences after discussion. To reduce problems for the parser, a sentence is processed as follows before being given to the preprocessor. First, some spoken forms are converted to written forms: *em* to *them*, *gonta* and *gon ta* to *going to*, *ta* to *to*, and *ya* to *you*. Second, the following tokens are removed: *eh*, *huh*, *oh*, *okay*, *uh*, *uhhuh*, *um*, *yeah*. Finally, if one or two words are repeated twice consecutively, e.g., *the the* or *I mean I mean*, the second occurrence is deleted. None of these steps changes the complexity of the input sentence.

Level	Training	Testing	Total
0	342	366	708
1	16	12	28
2	24	17	41
3	49	36	85
4	27	15	42
5	12	22	34
6	1	1	2
7	29	31	60
Total	500	500	1000

Table 1: Sentence distribution in the data

Level	Precision	Recall	F-Score
0	98.5%	95.6%	97.0%
1	100%	81.3%	89.7%
2	95.7%	91.7%	93.7%
3	78.3%	95.9%	86.2%
4	100%	88.9%	94.1%
5	84.6%	91.7%	88.0%
6	50.0%	100%	66.7%
7	87.9%	100%	93.6%
All	94.8%	94.8%	94.8%

Table 2: Results on the training data

Level	Precision	Recall	F-Score
0	100%	94.3%	97.1%
1	91.7%	91.7%	91.7%
2	92.9%	76.5%	83.9%
3	76.6%	100%	86.7%
4	76.5%	86.7%	81.3%
5	76.2%	72.7%	74.4%
6	33.3%	100%	50.0%
7	75.6%	100%	86.1%
All	93.2%	93.2%	93.2%

Table 3: Results on the test data

Level	0	1	2	3	4	5	6	7	Total
0	345	0	0	0	0	0	0	0	345
1	1	11	0	0	0	0	0	0	12
2	1	0	13	0	0	0	0	0	14
3	8	0	0	36	1	2	0	0	47
4	2	1	0	0	13	1	0	0	17
5	5	0	0	0	0	16	0	0	21
6	2	0	0	0	0	0	1	0	3
7	2	0	4	0	1	3	0	31	41
Total	366	12	17	36	15	22	1	31	500

Table 4: Confusion matrix for the test data

Experiment Results

Table 1 summarizes the distribution of the sentences in different levels in the training and test data. Given the young age of the children participating in the interactions, it is not surprising that the majority of the sentences in both the training and test data are simple sentences or sentence fragments in level 0.

Tables 2 and 3 summarize the results of the system on the training and test data for assigning sentences to different developmental levels. Precision, recall, and F-score for sentences in each level are computed as follows:

$$\text{Precision} = \frac{\text{Number of sentences correctly assigned to level}}{\text{Number of sentences assigned to level}} \quad (27)$$

$$\text{Recall} = \frac{\text{Number of sentences correctly assigned to level}}{\text{Number of sentences in level}} \quad (28)$$

$$\text{F-score} = \frac{2 * (\text{Precision} * \text{Recall})}{\text{Precision} + \text{Recall}} \quad (29)$$

The system achieves an overall accuracy of 94.8% and 93.2% on the training and test data respectively. A naive baseline model that assigns all sentences to level 0, the most likely assignment, would achieve an accuracy of 68.4% on the training data and 73.2% on the test data. The system’s performance constitutes an absolute improvement of 26.4% and 20% over that of the baseline model on the training and test data respectively.

In general, the system performs the best on sentences in levels 0, 1, 3 and 7, with F-scores of over 86% on both the training and test data. F-scores for sentences in levels 2, 4, and 5 range from high 80% to mid 90% on the training data, but drop to mid 70% to mid 80% on the test data. The reasons for the drop will be discussed in the error analysis section below. It is hard to draw any conclusive conclusion on the system’s performance on sentences in level 6, as the number of such sentences in both the training and test data is rather limited.

Error Analysis

Table 4 presents a confusion matrix for the scores assigned by the system to sentences in the test data. Each column in the table shows how many sentences in a given level are assigned to different levels. The matrix indicates two primary reasons for the system’s lower performance on sentences in levels 2, 4 and 5 in the test data than on those in the training data. First, a larger proportion of level-0 sentences are misscored in the test data, and the lower recall for level-0 sentences contributes to the lower precision for the system’s level-4 and level-5 assignments. Second, a larger proportion of sentences in the test data are misassigned to level 7, and the lower precision for the system’s level-7 assignments lead to lower recall for level-2 and level-5 sentences.

A close examination of the 34 erroneous assignments in the test data shows that all but one of them are caused by parsing errors that unavoidably lead the system to failure. Moreover, four major sources of parsing errors are identified. First, 11 of these utterances contain two or more sentences joined with no punctuation or conjunction, causing the parser to arrive at erroneous complex structures.

An example is *no we didn't that's right*. In this case, the parser analyzes *that's right* as an SBAR child of the VP headed by *did*. This causes the heuristic in (15) to fire and assign the utterance to level 5. However, such utterances are assigned to level 0 by the human annotators, because there is no conjunction between the two sentences in the utterance. For complexity analysis, such utterances should probably be treated as two sentences by the transcribers. Second, another 10 of the 34 utterances contain extra elements such as false starts, repetitions, corrections, etc., leading to misanalysis by the parser. An example of this is *you're spike is somebody's else Ross right?* In this case, the parser treats *spike is somebody's else Ross right* as an SBAR-A child of the VP headed by *'re*, causing the heuristic in (19) to fire and assign the utterance to level 3. Third, 5 other parsing errors are caused by the absence of punctuation between the main sentence and summons (e.g., *daddy*) or informal colloquial forms such as *come on*. For example, in *come on you can be articulate*, the parser misanalyses *come* as the main verb and the rest of the sentence as a subordinate clause. All of the three types of errors demonstrate the special challenges spoken language poses to the parser. Finally, the other 7 parsing errors are due to the failure of the parser or the tagger to correctly handle structural or POS ambiguity or process a long utterance. For example, in the utterance *somebody or it's raining outside*, the parser analyzes *somebody or it* as a conjoined subject NP, causing the heuristic in (13) to fire and assign the utterance to level 2.

Discussion and Conclusion

This paper described a heuristics-based system for automatic syntactic complexity measurement using the revised D-Level Scale. The system should prove useful to researchers in a variety of language-related fields whose work involves applying the D-Level Scale to large language samples. Experiment results show that the system achieves an accuracy of 93.2% on unseen child language acquisition data from the CHILDES database. Close analysis of the system's erroneous assignments indicate that the heuristics developed are highly effective and that misassignments are caused primarily by parsing errors. A number of challenges spoken language poses to the parser are identified. Future research should explore ways to normalize utterances in spoken data to reduce problems for the tagger and parser.

References

Cheung, H. and Kemper, S. (1992). Competing complexity metrics and adults' production of complex sentences. *Applied Psycholinguistics* 13, 53-76.

Collins, M. (1999). *Head-Driven Statistical Models for Natural Language Parsing*. Ph.D. dissertation, University of Pennsylvania, Philadelphia, PA.

Covington, M. A., He, C., Brown, C., Naci, L., and Brown, J. (2006). How complex is that sentence? A proposed revision of the Rosenberg and Abbeduto D-Level Scale. CASPR Research Report 2006-01. Athens, GA: The University of Georgia, Artificial Intelligence Center.

Graesser, A., McNamara, D. S., Louwerse, M., and Cai, Z. (2004). Coh-Metrix: Analysis of text on cohesion and language. *Behavioral Research Methods, Instruments, and Computers* 36, 193-202.

Lee, L. (1974). *Developmental Sentence Analysis*. Evanston, IL: Northwestern University Press.

Long, S. H., Fey, M. E., and Channell, R. W. (2004). Computerized Profiling (Version 9.6.0). Cleveland, OH: Case Western Reserve University.

Macleod, C., Grishman, R., Meyers, A., Barrett, L., and Reeves, R. (1998). NOMLEX: A lexicon of nominalizations. In *Proceedings of the 8th International Congress of the European Association for Lexicography*.

MacWhinney, B. (2000). *The CHILDES Project: Tools for Analyzing Talk*. Mahwah, NJ: Lawrence Erlbaum Associates.

Marcus, M. P., Santorini, B., and Marcinkiewics, M. A. (1993). Building a large annotated corpus of English: The Penn Treebank. *Computational Linguistics* 19(2), 313-330.

Ortega, L. (2003). Syntactic complexity measures and their relationship to L2 proficiency: A research synthesis of college-level L2 writing. *Applied Linguistics* 24(4), 492-518.

Rosenberg, S., and Abbeduto, L. (1987). Indicators of linguistic competence in the peer group conversational behavior of mildly retarded adults. *Applied Psycholinguistics* 8, 19-32.

Sagae, K., Lavie, A., and MacWhinney, B. (2005). Automatic measurement of syntactic development in child language. In *Proceedings of ACL-05*, 197-204.

Scarborough, H. S. (1990). Index of Productive Syntax. *Applied Psycholinguistics* 11, 1-22.

Snowdon, D. A., Kemper, S. J., Mortimer, J. A., Greiner, L. H., Wekstein, D. R., and Markesbery, W. R. (1996). Linguistic ability in early life and cognitive function and Alzheimer's disease in late life: Findings from the Nun Study. *Journal of the American Medical Association* 277(7), 528-532.

Tsuruoka Y. and Tsujii, J. (2005). Bidirectional inference with the easiest-first strategy for tagging sequence data. In *Proceedings of HLT/EMNLP-05*, 467-474.