Representations for category disambiguation

Markus Dickinson
Indiana University
Bloomington, IN 47405
md7@indiana.edu

Abstract

As it serves as a basis for POS tagging, category induction, and human category acquisition, we investigate the information needed to disambiguate a word in a local context, when using corpus categories. Specifically, we increase the recall of an error detection method by abstracting the word to be disambiguated to a representation containing information about some of its inherent properties, namely the set of categories it can potentially have. This work thus provides insights into the relation of corpus categories to categories derived from local contexts.

1 Introduction and Motivation

Category induction techniques generally rely on local contexts, i.e., surrounding words, to cluster word types together (e.g., Clark, 2003; Schütze, 1995), using information of a kind also found in human category acquisition tasks (e.g., Mintz, 2002, 2003). Such information is also at the core of standard part-of-speech (POS) tagging, or disambiguation, methods (see, e.g., Manning and Schütze, 1999, ch. 10), with the contexts generally abstracted to POS tags. The contextual information is similar in both tasks because induction is founded in part upon the notion that local contexts are useful for disambiguation: one morphosyntactically clusters words which should have the same category in the same contexts. But which contexts count as being the "same"? And to what extent do categories based on context distributions resemble corpus annotation categories? Since disambiguation is in some sense more primary, to begin to answer these questions we investigate which representations are effective for category disambiguation.

Disambiguating a word's category in context has of course been explored in other situations, especially POS tagging. Rarely, however, has it been shown as to which information is the most accurate at disambiguation and which information is absolutely necessary, without mixing these issues with other tagging issues, such as smoothing and unknown word tagging. We need techniques which isolate disambiguation, placing less emphasis on generalizing contexts to new data. To determine the essential information needed for accurate disambiguation, we start with a precise model and generalize it. Changing the model in small ways and evaluating the resulting precision will indicate how particular aspects of the representation are contributing to successful disambiguation.

The central question of this paper is: which representation (of a word and its context) indicates that two situations should be categorized the same? In this context, POS annotation error detection provides an ideal setting to explore representations for disambiguation. Error detection relies on the assumption that words should be annotated consistently—in other words, contexts are grouped which accurately identify the category of a word as being consistent—and it does this with an emphasis on high precision. In essence, error detection already investigates where disambiguation can be done, often using local contexts (e.g., Dickinson, 2005). With an emphasis on high precision, however, many corpus instances are essentially uncategorized and are thus in need of generalization.

To get at the central question of an appropriate

Proceedings of the 22nd International Conference on Computational Linguistics (Coling 2008), pages 201–208
Manchester, August 2008

representation for disambiguation, then, our task is to generalize error detection and increase the recall of errors found in a corpus by exploiting more general properties of a corpus. Given that annotation errors can have a profound impact on the quality of training and testing on such data (see Dickinson, 2005, ch. 1), this task also serves an immense practical need in its own right.

In exploring error detection recall, we can connect the task to another with much of the same emphasis. Human category acquisition experiments have also focused on precision: instead of asking how every word is categorized, they examine how some words are categorized, from which others can be bootstrapped. As outlined in sections 2 and 3, we can use such studies as a starting point for generalizing error detection.

2 Background

2.1 The variation n-gram method

The error detection method we build from is the variation n-gram method (Dickinson and Meurers, 2003; Dickinson, 2005). The approach detects items which occur multiple times in the corpus with varying annotation, the so-called *variation nuclei*. A nucleus with its repeated surrounding context is referred to as a *variation n-gram*. Every detected variation in the annotation of a nucleus is classified as an error or a genuine ambiguity using a basic heuristic requiring at least one word of context on each side of the nucleus.

For example, in the WSJ corpus, part of the Penn Treebank 3 release (Marcus et al., 1993), the string in (1) is a variation 12-gram since *off* is a variation nucleus that is tagged preposition (IN) in one corpus occurrence and particle (RP) in another.[1] Dickinson (2005) shows that examining those cases with identical local context—in this case, looking at *ward off a*—results in an estimated error detection precision of 92.5%.

(1) to ward off a hostile takeover attempt by two European shipping concerns

This method can be applied to syntactic annotation, and for this annotation, one can increase the recall of errors found by abstracting the nuclei to POS tags (Boyd et al., 2007). Clearly, this is not a feasible abstraction here, given that we are attempting to detect errors in POS annotation.

[1]To distinguish variation nuclei, we shade them in gray and underline the immediately surrounding context.

2.2 Frames for language acquisition

Research on language acquisition has addressed the question of how humans discover and learn categories of words, using virtually the same contexts as in the variation n-gram method. Mintz (2002) shows that local context, in the form of a *frame* of two words surrounding a target word, leads to categorization in adults of the target, and Mintz (2003) shows that frequent frames supply category information, consistent across child language corpora. A frame is defined as "two jointly occurring words with one word intervening" (Mintz, 2003), e.g., *you __ it*. The frame is not decomposed into its left side and right side (cf., e.g., Redington et al., 1998; Clark, 2003) , but instead is taken as the occurrence of both sides. The *target word* is the intervening word, but it is not included in the frame (unlike variation nuclei).

For category acquisition, only *frequent frames* are used, those with a frequency above a certain threshold. Frequent frames predict category membership: the set of words appearing in a given frame should represent a single category. The frequent frame *you __ it*, for example, largely identifies verbs, as shown in (2), taken from the CHILDES database of child-directed speech (MacWhinney, 2000). Analyzing the frequent frames in six subcorpora of CHILDES, Mintz (2003) obtains both high type and high token accuracy in grouping words into the same categories.

(2) a. you put it
b. you see it

To take this work as a basis for investigating disambiguation, some points are in order about the results. First, accuracies slightly degrade when moving from the "Standard Labeling" category set[2] to the more fine-grained "Expanded Labeling" category set,[3] i.e., a .98 to .91 drop in token accuracy and .93 to .91 drop in type accuracy. It is not clear what happens with even more fine-grained corpus tagsets. Secondly, Mintz (2003) assumes that, at least for his experiments, each word has only one class (see also Redington et al., 1998, p. 439-440). The tasks of category induction and category disambiguation are thus conflated into a single step. We do not know for sure whether frames induce

[2]Categories = noun, verb, adjective, preposition, adverb, determiner, *wh*-word, *not*, conjunction, and interjection.

[3]Nouns split into nouns and pronouns; verbs split into verbs, auxiliaries, and copula

coherent sets of words or whether they accurately disambiguate a word, or both. In other words, can frames be used to group the target words (induction) or to group the contexts (disambiguation)?

While we investigate using frames for disambiguation in English (and somewhat in German), the concept of a frame has been shown to be cross-linguistically viable (Chemla et al., in press), and in principle could extend to languages encoding relations through morphology instead of linear order (see the discussion in Mintz, 2003).

3 Generalizing error detection via frames

Both strands of research employ local contexts for identifying categories, but the variation n-gram method relies on identical words to serve as variation nuclei, or target words to be disambiguated. To increase the recall of the method in a way relating to acquisition, the nucleus should be abstracted to something more general than a word. As a (frequent) frame does not include the target, predicting that the category within that context is always the same, a first step in abstracting the nucleus is to require no similarity between nuclei.

We thus search for all identical nuclei with frame context—or what we will call *framed variation nuclei*—such that there is variation in labeling for the nucleus, but we require no identity of the nucleus. We investigate the WSJ portion of the Penn Treebank, and, to provide more robust evaluation, also compare the TIGER corpus of German, version 2 (Brants et al., 2002) where appropriate. Given that punctuation is less informative for determining a category, we remove from consideration frames containing punctuation as one of the context words, and obtain 48,717 variations in the WSJ and 22,613 in TIGER.

Although basic hand-examination reveals some errors, a majority of cases contain acceptable variations. As one example, in the WSJ the frame *the* __ *of* occurs as the most frequent frame with variation in labeling for the target (5737 instances). This is a nominal position, and thus we find variation between a variety of correct nominal tags: cardinal number (CD), adjective (JJ, JJR, JJS), common noun (NN, NNS), and proper noun (NNP, NNPS), in addition to the erroneous verbal tags VBD (past tense verb) and VBG (verb, *-ing* form). Restricting our attention to the frequent frames, as in Mintz (2003), is not helpful: the problem occurs irrespective of frequency. Indeed, there is an average of 2.56 categories per variation, with one variation (*and __ in*) having 21 categories. This is consistent in TIGER, which has 2.57 categories per variation and 22 categories for *und __ in*.

While more context could help, the real issue is the definition of a nucleus. In the example above, which nominal tag is used depends upon inherent properties of the word involved. Consider the frame *that __ the*. Among the 18 possible tags, there is variation between NN (common noun) for words like *afternoon* and VBZ (present tense verb, 3rd person singular) for words like *says*. Both are legitimate, and the primary way to tell is by examining information about the target word. In generalizing the nucleus, instead of abstracting it to nothing, we need to abstract it to something indicating broad characteristics of the word.

4 An appropriate level of abstraction

On the one hand, the variation n-gram method has high precision; on the other, using frames results in high recall, but too low a precision to sort through. Both methods rely on the same identical contexts; the issue is in finding which words are comparable. Consider the frame *n't __ that*. Some words are inherently similar and should have the same tags: the correct *n't help/VB that* and the erroneous *n't matter/NN that*, for instance, are comparable. Other cases are not: *one/CD* and *shown/VBN* can never have the same category. We need to find classes of words that, within the same context, should not vary in their annotation, and it makes sense to compare words in context if they have the same category possibilities.

4.1 Complete ambiguity classes

Ambiguity classes capture the relevant property we are interested in: words with the same category possibilities are grouped together.[4] And ambiguity classes have been shown to be successfully employed, in a variety of ways, to improve POS tagging (e.g., Cutting et al., 1992; Daelemans et al., 1996; Dickinson, 2007; Goldberg et al., 2008; Tseng et al., 2005). Only certain words can take one of two (or more) tags, and these should be disambiguated in the same way in context. As an example of how using ambiguity classes as variation nuclei can increase recall, consider the frame *being __ by* in example (3). There are at least 27

[4] One could group affixes by ambiguity class for languages like Chinese (cf. CTBMorph features in Tseng et al., 2005).

different VBN (past participle) verbs appearing between *being* and *by* (3a), but none of these verbs ever appear as VBD here, even though all of them could be VBD. Two other VBD/VBN verbs, *rejected* (3b) and *played* (3c), erroneously appear as VBD here, but never as VBN. With the nucleus VBD/VBN, we can find this erroneous variation.

(3) a. being { raised/VBN , infringed/VBN , supported/VBN , ... } by

b. as probable as being rejected/VBD by the Book-of-the-Month Club

c. the ... role in takeover financing being played/VBD by Japanese banks

Thus, to define *complete ambiguity class* variation nuclei, we make a first pass through the corpus to calculate every word's ambiguity class. On a second pass, the ambiguity class serves as the (framed) variation nucleus, e.g., *being VBD/VBN by*. Ambiguity class nuclei with more than one tag in a frame context are flagged as a potential error.

4.2 Pairwise ambiguity classes

While abstracting to a word's possible classes can increase the number of errors found, potentially erroneous classes prevent further increased recall. For example, the class for *plans* is erroneously classified as NNS/VBP/VBZ, even though its one instance of VBP (present tense verb, non-3rd person singular) in the corpus is erroneous. Without that case, we would have NNS/VBZ and more directly comparable words.

As a second experiment, then, we define *pairwise ambiguity class* variation nuclei, using subsets of ambiguity classes to define a nucleus. If the variation is only between NNS and VBZ, we need to allow all words with NNS/VBZ variation to count as comparable nuclei. As above, we calculated a word's ambiguity class during a first pass. In the second pass through the corpus, we break the ambiguity class down into its pairs, and each relevant pair is stored as a variation nucleus. The relevant pairs of tags are those which contain the tag at that position since classes without that tag can never have meaningful variation. Taking the example of *company plans to*, with the ambiguity class NNS/VBP/VBZ for *plans*, if the current corpus position marks *plans* as NNS, then we store the two trigrams in (4).

(4) a. company NNS/VBZ to

b. company NNS/VBP to

Looking over the whole corpus, we find variation between NNS and VBZ, but none between NNS and VBP. In principle, this instance of *plans/NNS* could be in both an NNS/VBZ and an NNS/VBP variation; this is necessary since we do not a priori know which variations will be problematic.

5 Results and Insights

5.1 Complete ambiguity classes

Using complete ambiguity class variation nuclei, we find 4131 framed variation nuclei in the WSJ. Almost all variations involve only two or three tags, with 2.03 tags per variation. TIGER has 626 framed variation nuclei, with 2.01 tags per variation.

From the 4131 variations, we randomly sampled 100 cases and hand-evaluated whether they contain an error, and whether its detection is attributable to the generalization to complete ambiguity classes. Of the 100, 79 of the cases contain at least one error, and 15 of these cases are new examples, i.e., cases without identical words. With a point estimate of .79, we estimate 3263 errors and obtain a 95% confidence interval of (0.7102, 0.8698), meaning that we predict between 2933 and 3593 of the 4131 cases contain errors. The 79 erroneous cases point to 134 token errors, of which 23 are new.

In addition to increasing the recall of the method, the cases are arguably more thoroughly grouped than before. For instance, we see in (5) that both *pretax* and *third-quarter* vary between JJ and NN in the variation *said JJ/NN profit*, with *first-half* additionally appearing only as JJ. Since JJ is the correct tag for all instances, the two NN errors are detected with word nuclei, but here all the relevant examples are together. This provides evidence for the claim that an ambiguity class is a level of abstraction supporting identical disambiguation in the same context.

(5) said { first-half/JJ , third-quarter/JJ , pretax/JJ , third-quarter/NN , pretax/NN } profit

The recall has increased, but 79% is below the 92.5% precision previously obtained for the variation n-gram method with word nuclei (Dickinson, 2005). However, that result used *distinct* variation

nuclei, meaning that the longest contexts were examined before working down to shorter contexts. Furthermore, it is not clear how well the original word nuclei method scales up to larger corpora. Some of the new false positives we observe would likely be false positives for word nuclei, given more data. For example, the new method turns up *generally VBD/VBN the* as a false positive, as in (6), because of the non-local tagset distinction and short context. With more data, we are more likely to see an acceptable use of, e.g., *generally favored/VBD the*, a false positive for word nuclei. In some sense, then, this 79% precision might be a more general indication of the method's precision for this tagset and genre.

(6) a. TV news coverage has generally favored/VBN the government
b. Members ... generally received/VBD the regional officials

Finally, of the 21 false positives (20 of which are new), five of them stem from an error in the ambiguity class, corresponding to five token errors. For example, there is variation for JJ/NN words in the frame *of __ pills*, as in (7). However, *poison* should never be JJ: its ambiguity class should be NN, not the incorrect JJ/NN. For error detection, this means 84 of the 100 samples lead to some kind of POS error; for investigating disambiguation contexts, this means that 83% (79/95) of the cases support complete disambiguation. Thus, when abstracting to ambiguity class nuclei, local context generally provides sufficient information for disambiguation (see also section 6).

(7) of { birth-control/JJ , poison/NN } pills

One limitation of the variation n-gram method is the fact that some distinctions often need non-local information (cf. (6)). A bigger problem for grouping words by ambiguity classes is the fact that annotation can be semantically-based. For example, the variation *of JJ/NN bank* is a legitimate ambiguity because the distinction between JJ and NN is semantic. Compare *a sort of merchant/NN bank* with *an extension of senior/JJ bank debt*: both nuclei are clearly in a noun modifier position, but the tags are different based on what they denote. This shows the limitations of local distributional information without lexical information, for making these tagset distinctions.

5.2 Pairwise ambiguity classes

With pairwise ambiguity classes serving as variation nuclei, we find 6235 variation frames in the WSJ and 874 in TIGER, significant increases over using complete ambiguity class nuclei. To evaluate the method, we want to know: a) how many total errors we detect, b) how many of these were detected by using either complete or pairwise ambiguity classes, and c) how many were detected specifically with pairwise ambiguity classes.

A sample of 100 of the WSJ cases reveals (a) 59 total errors, (b) 18 of which involve ambiguity class nuclei that would not have been found with word nuclei. Of these 18, (c) 8 cases can only be found by extending the method to pairwise classes. For the point estimate of .59, we estimate approximately 3679 variations to be errors (95% CI: 3078 to 4280 errors). The 59 erroneous variations point to a total of (a) 134 token errors, (b) 30 of which were detected by ambiguity classes; (c) 17 of these were detected by pairwise ambiguity classes. Clearly, using pairwise ambiguity classes increases the number of errors found.

As an example, consider (8), centering on the frame *came __ for*. The original variation n-gram method turns up no variation here, but neither does the complete ambiguity class extension: *in* has the ambiguity class FW/IN/NN/RB/RBR/RP, and *out* the class IN/JJ/NN/RB/RP. Since the only relevant variation is between IN and RP, the pairwise nuclei method turns up such cases with the variation *came IN/RP for*, pointing to an error in the two cases of *out*.

(8) a. accounts came in/RP for some blocks
b. numbers came out/IN for September
c. he again came out/IN for an amendment

But what of the 41 false positives, 22 of which are due to the pairwise classes? We have increased recall, but there is also a 20% absolute drop in precision. Is this tradeoff worth it? To answer this, it is important to note that 15 of the false positives are due to faulty ambiguity classes, as discussed above, and 10 of those 15 are from pairwise classes. For error detection, this means 74 of the 100 samples lead to some POS error; for investigating disambiguation contexts, this means 69% (59/85) of the cases support disambiguation.

Additionally, the 15 cases point to 53 token errors, much more than in the previous experiment, due to 44 token errors from the new pair-

wise ambiguity classes. For example, in the variation frame *as DT/JJ sales*, the words which vary are *a* (tagged DT (determiner), with a complete ambiguity class of DT/FW/IN/JJ/LS/NNP/SYM) and *many* (tagged JJ, with an ambiguity class of DT/JJ/NNS/PDT/RB/VB). Unsurprisingly, *a* should never have been tagged JJ in the corpus, i.e., its ambiguity class is wrong.

In addition to the issue of erroneous tags in an ambiguity class, atypical tags also pose a problem. Consider the frame *that JJ/RB in*, as illustrated in (9), with acceptable variation. It might appear that *sometime* has a problem with its ambiguity class, but the use of JJ is actually correct, as shown in (10), where *sometime* is atypically modifying a noun. To counter atypical uses, one could use only "typical" ambiguity classes (cf. Dickinson, 2007) or define ambiguity classes according to order of frequency (cf. Daelemans et al., 1996), e.g., JJ/RB vs. RB/JJ.

(9) a. a departure from the past that many/JJ in the industry ...
    b. hope that sometime/RB in the near future

(10) real estate magnate and sometime/JJ raider Donald Trump

This illustrates that the selection of an abstracted class for a nucleus definition is non-trivial, and ambiguity classes are simply an approximation.

POS contexts One problem for our method is that word contexts are not always truly comparable; identical context words can be used differently. For instance, with the variation *that NN/VBP along* in (11), the uses of *that* are clearly distinct and are marked as such by their tags.

(11) a. gifts that/WDT go/VBP along with purchases
    b. We are considering that/DT offer/NN along with all other alternatives

But do tagset categories actually aid in local disambiguation? To quickly gauge this, we take the previous sample of 100 variations and recover the POS information for the context. Isolating those cases with non-identical POS tags for the same word contexts, we find 10 examples and hypothesize that these will more likely be acceptable variations. Interestingly, however, of those ten, six successfully identified errors; it turns out that the POS of the word is often irrelevant for disambiguation. For the variation *paid JJR/RBR than* in (12), for example, the tag of the context word *paid* is different in these cases, but that does not matter for the tag of *more*, which should be consistent.

(12) a. they paid/VBD more/JJR than $ 1 million
    b. he has paid/VBN more/RBR than $ 70,000

More problematically, four of the erroneous variation nuclei also contained POS errors in the context, as in example (13). The variation *all CC/RB disappeared* points to an error in the word *but*, yet there is also a noticeable inconsistency in the word *all*.

(13) a. have all/DT but/CC disappeared
    b. have all/RB but/RB disappeared

In other words, it is often the case that we should ignore the POS of the context words, due to the fact that erroneous contexts exist and, more importantly, that not all categories aid in disambiguation. Exploring which contextual categories aid in target category disambiguation (cf., e.g., Brants, 1997) could aid in developing better disambiguation models, and perhaps also a better sense of what categories are useful to induce (e.g., a broader category *Verb* in (12) for *paid*).

6 Representations for disambiguation

We have shown that local lexical context provides a generally unambiguous context for corpus tags, given sufficient information about the word to be disambiguated. The information need not be very abstract, either: frames using ambiguity class nuclei only require a word's category possibilities. Even for many unsupervised situations, this is available from a lexicon (e.g., Banko and Moore, 2004; Goldberg et al., 2008).

We have only looked at cases with variation in tagging; fully gauging the accuracy of such a data representation for disambiguation requires more of the framed nuclei from the corpus, including those without variation. For this, we could take all framed nuclei from a corpus and compare the level of ambiguity for differing abstractions. However, most framed nuclei occur only once, and it is not clear how meaningful it is to say that these are unambiguous. Thus, we examine framed nuclei which occur at least twice and report in table 1

for the WSJ how unambiguous a particular level of nucleus abstraction is.[5]

Abstraction	Unamb.	Total	Accuracy
Word	84,784	87,390	97.02%
Complete AC	90,341	94,472	95.63%
No info.	51,945	100,662	51.60%

Table 1: Disambiguation accuracy for the WSJ

While abstracting to the case where the nucleus contains no information (*No info.*) creates more cases which are classifiable—over 100,000—the accuracy of disambiguation drops from the upper 90% range to 52%. Note, however, that the abstraction to complete ambiguity class (AC) nuclei has minimal degradation in accuracy, yet increases the number of accurately classified cases. When we recall that approximately 79% of of the 4131 variation frames should have a single tag, i.e., 3263 cases, this means that the overall disambiguation accuracy is estimated to be 99.08% (93,604/94,472).

In addition to the disambiguation accuracy of frames, we can look at the accuracy of word tokens identified by frames. To gauge this, we identify the most likely tag of each framed variation nucleus and assign it to all instances of the nucleus. In the case of ties, one tag is randomly selected; since we are only calculating overall word token accuracy, the exact tag selected is unimportant. The results of comparing to the benchmark tags are given in table 2. Even though the abstraction to no information identifies more word tokens, the ambiguity class abstraction correctly categorizes nearly as many words.

Abstraction	Correct	Total	Accuracy
Word	340,860	345,139	98.76%
Complete AC	441,603	448,402	98.74%
No info.	444,635	582,601	76.32%

Table 2: Word token accuracy for the WSJ

With the smaller and likely more accurately tagged TIGER corpus, we find exactly the same trends, as shown in table 3. This supports the claim across corpora that local context is often sufficient to disambiguate a word, if some information from the word—here, the category possibilities—is present in the nucleus.

[5]As pairwise ambiguity classes involve more than one nucleus per corpus position, we use complete ambiguity classes.

Abstraction	Unamb.	Total	Accuracy
Word	37,038	37,324	99.23%
Complete AC	47,832	48,458	98.71%
No info.	33,881	56,494	59.97%

Table 3: Disambiguation accuracy for TIGER

The poor accuracy for framed nuclei with no information indicates that methods which intend to match corpus annotation categories could face difficulties in obtaining a single category without using more information. There is still much space to explore, however, between using ambiguity class nuclei and no information, in order to further increase the number of comparable cases without losing accuracy and in order to be more knowledge-free.

7 Summary and Outlook

Motivated by work on category acquisition, we have shown that local contexts—i.e., immediately surrounding words, or frames—can delineate corpus categories when the level of abstraction for the word to be disambiguated indicates some inherent properties of the word, namely the categories it can have. By abstracting away from lexical items to broader classes of words, we have been able to increase the recall of an error detection method without much drop in its precision.

Having successfully defined a representation for disambiguation, the next step is to make the representation more general, in order to include more comparable instances. As what we have done is essentially a form of nearest neighbor classification, one could in the future explore more sophisticated techniques to cluster contexts.

At the same time, we wish to use as little annotated knowledge as possible. Thus, an orthogonal line of research can involve inducing classes for words which are more general than single categories, i.e., something akin to ambiguity classes (see, e.g., the discussion of ambiguity class guessers in Goldberg et al., 2008). This could make error detection completely independent of the annotation and, more importantly, lead to an improved understanding of the best knowledge-free representation for disambiguation.

Since induction is founded to some extent upon disambiguating contexts, this work has some bearing on the evaluation of induced categories with corpus annotation; not only is there more than

one tagset in existence (see discussion in Clark, 2003), but annotation schemes make distinctions that morphosyntactic contexts cannot readily capture. For example, there is an implicit notion of inherency in the distinction between JJ and NN in the Penn Treebank (Santorini, 1990, p. 12-13). Fully outlining these inherent properties could provide insights into induction and its evaluation.

Acknowledgments

Thanks to the three anonymous reviewers for their useful comments and to Charles Jochim for helpful discussion. This material is based upon work supported by the National Science Foundation under Grant No. IIS-0623837.

References

Banko, Michele and Robert C. Moore (2004). Part-of-Speech Tagging in Context. In *Proceedings of COLING 2004*. Geneva, Switzerland, pp. 556–561.

Boyd, Adriane, Markus Dickinson and Detmar Meurers (2007). Increasing the Recall of Corpus Annotation Error Detection. In *Proceedings of TLT 2007*. Bergen, Norway, pp. 19–30.

Brants, Sabine, Stefanie Dipper, Silvia Hansen, Wolfgang Lezius and George Smith (2002). The TIGER Treebank. In *Proceedings of TLT-02*. Sozopol, Bulgaria.

Brants, Thorsten (1997). Internal and External Tagsets in Part-of-Speech Tagging. In *Proceedings of Eurospeech*. Rhodes, Greece.

Chemla, E., T. H. Mintz, S. Bernal and A. Christophe (in press). Categorizing words using 'Frequent Frames': What cross-linguisic analyses reveal about core principles. *Developmental Science* .

Clark, Alexander (2003). Combining Distributional and Morphological Information for Part of Speech Induction. In *Proceedings of EACL-03*. Budapest, pp. 59–66.

Cutting, Doug, Julian Kupiec, Jan Pedersen and Penelope Sibun (1992). A Practical part-of-speech tagger. In *Proceedings of the ANLP-92*. Trento, Italy, pp. 133–140.

Daelemans, Walter, Jakub Zavrel, Peter Berck and Steven Gillis (1996). MBT: A Memory-Based Part of Speech Tagger-Generator. In *Proceedings of the Fourth Workshop on Very Large Corpora (VLC)*. Copenhagen, pp. 14–27.

Dickinson, Markus (2005). Error detection and correction in annotated corpora. Ph.D. thesis, The Ohio State University.

Dickinson, Markus (2007). Determining Ambiguity Classes for Part-of-Speech Tagging. In *Proceedings of RANLP-07*. Borovets, Bulgaria.

Dickinson, Markus and W. Detmar Meurers (2003). Detecting Errors in Part-of-Speech Annotation. In *Proceedings of EACL-03*. Budapest, pp. 107–114.

Goldberg, Yoav, Meni Adler and Michael Elhadad (2008). EM Can Find Pretty Good HMM POS-Taggers (When Given a Good Start). In *Proceedings of ACL-08*. Columbus, OH, pp. 746–754.

MacWhinney, Brian (2000). *The CHILDES project: Tools for analyzing talk*. Mahwah, NJ: Lawrence Erlbaum Associates, third edn.

Manning, Christopher D. and Hinrich Schütze (1999). *Foundations of Statistical Natural Language Processing*. Cambridge, MA: The MIT Press.

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

Mintz, Toben H. (2002). Category induction from distributional cues in an artificial language. *Memory & Cognition* 30, 678–686.

Mintz, Toben H. (2003). Frequent frames as a cue for grammatical categories in child directed speech. *Cognition* 90, 91–117.

Redington, Martin, Nick Chater and Steven Finch (1998). Distributional Information: A Powerful Cue for Acquiring Syntactic Categories. *Cognitive Science* 22(4), 425–469.

Santorini, Beatrice (1990). *Part-Of-Speech Tagging Guidelines for the Penn Treebank Project (3rd Revision, 2nd printing)*. Tech. Rep. MS-CIS-90-47, The University of Pennsylvania, Philadelphia, PA.

Schütze, Hinrich (1995). Distributional Part-of-Speech Tagging. In *Proceedings of EACL-95*. Dublin, Ireland, pp. 141–148.

Tseng, Huihsin, Daniel Jurafsky and Christopher Manning (2005). Morphological features help POS tagging of unknown words across language varieties. In *Proceedings of the Fourth SIGHAN Workshop on Chinese Language Processing*.