A Challenge Set and Methods for Noun-Verb Ambiguity

Ali Elkahky, Kellie Webster, Daniel Andor, and Emily Pitler
Google AI Language
{alielkahky, websterk, andor, epitler}@google.com

Abstract

English part-of-speech taggers regularly make egregious errors related to noun-verb ambiguity, despite having achieved 97%+ accuracy on the WSJ Penn Treebank since 2002. These mistakes have been difficult to quantify and make taggers less useful to downstream tasks such as translation and text-to-speech synthesis. This paper creates a new dataset of over 30,000 naturally-occurring non-trivial examples of noun-verb ambiguity. Taggers within 1% of each other when measured on the WSJ have accuracies ranging from 57% to 75% accuracy on this challenge set. Enhancing the strongest existing tagger with contextual word embeddings and targeted training data improves its accuracy to 89%, a 14% absolute (52% relative) improvement. Downstream, using just this enhanced tagger yields a 28% reduction in error over the prior best learned model for homograph disambiguation for text-to-speech synthesis.

1 Introduction

Whether a word is functioning as a noun or a verb in a particular linguistic context critically affects the output of tasks including translation and text-to-speech synthesis. The English word *close* may be translated as either *nah* (adjective/non-verb) or *schließen* (verb) (example from Sennrich and Haddow (2016)). In text-to-speech, the homograph *lives* is pronounced /laɪvz/ (noun) or /lɪvz/ (verb; example from Sproat et al. (1992)).

While downstream applications require taggers be sensitive to non-local linguistic context, it is difficult to measure such sensitivity with current tagging evaluation. In the past 15 years since Collins (2002), many models have accuracy exceeding 97% when measured on the WSJ Penn Treebank, which is within the level of human inter-annotator agreement for the corpus. Incorporating non-local context via sentence-based representations (Collobert et al., 2011) or state-of-the-art contextual representations of tokens (ELMo, Peters et al. (2018)) yields the same tagging accuracy as Collobert et al.'s limited window-based representation (97.3%). However, existing local models "regularly make egregious errors" (Manning, 2011), notably on imperative detection[1]. That is, the applicability of the part-of-speech labeling task is limited by its standard evaluation not reflecting difficult cases which require contextual reasoning to resolve ambiguity.

In this paper, we address this mismatch by creating a targeted intrinsic evaluation: a challenge dataset of over 30,000 naturally-occurring non-trivial examples of noun-verb ambiguity spanning multiple domains and containing many imperatives that non-expert humans can annotate with high agreement (Section 2). We will publicly release both the training and evaluation data[2].

We further contribute a series of modeling experiments on this data. We first show that state-of-the-art taggers perform poorly on this challenge (Table 1) and then investigate two simple and orthogonal approaches to enhancing a state-of-the-art tagger: incorporating generic contextual embeddings trained on billions of words, and incorporating thousands of examples of training data targeted for this task. Both of these approaches yield large and complementary improvements: the combined methods give an accuracy of 89.1%, a 14% absolute improvement over a state-of-the-art tagger and a 31% absolute improvement over the widely used Stanford tagger. Section 3 provides an overview of the investigated taggers, experiments, and results.

[1] In experiments with recipe data in Kiddon et al. (2015), an unsupervised system had an F1 score over 20% higher in absolute terms than supervised taggers.

[2] http://goo.gl/language/noun-verb

Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing, pages 2562–2572
Brussels, Belgium, October 31 - November 4, 2018.

Model	WSJ	NV
Existing Taggers		
Toutanova et al. (2003)	97.24	57.6
Choi (2016)	97.64	71.2
Dozat et al. (2017)	97.33	70.4
Bohnet et al. (2018)	**98.00**±.12	74.0±1.2
Enhancements		
+ELMo	**97.94**±.08	82.1±0.9
+NV Data	**97.98**±.11	86.4±0.4
+ELMo+NV Data	**97.97**±.09	**88.9**±0.3

Table 1: Empirical Results. All investigated new and existing taggers are within 1% of each other when measured on the WSJ test set. When evaluated on the Noun-Verb dataset, however, existing taggers range from 57% to 74%. Adding enhancements to the Bohnet et al. (2018) tagger gives over 14% absolute improvement. Best results and results insignificantly different from the best are bolded (two-tailed t-test).

Finally, we demonstrate that these tagging improvements make a positive impact on the downstream task of homograph disambiguation for text-to-speech (Section 4).

2 Noun-Verb Dataset

Consider the ambiguous examples below:

(1) Certain insects can damage plumerias, such as mites, **flies**, or aphids. **NOUN**

(2) **Mark** which area you want to distress. **VERB**

All tested existing part-of-speech taggers (Table 1) mistag both of these examples, tagging *flies* as a verb and *Mark* as a noun[3]. Looking at only the WSJ Penn Treebank, all occurrences of *Mark* are nouns, so a part-of-speech tagger that ignores context completely could appear to do quite well on this word type. Similarly, all occurrences of the word type *share* in the WSJ development set are noun instances.

A baseline of selecting the most frequent tag per word type (ignoring all context) achieves 93.0% accuracy on the ambiguous tokens in the WSJ (Table 2). A simple tagger based on a single hidden layer feed-forward neural network with 128 units that uses a three word window around the focus token as features achieves an accuracy of 97.0% on the WSJ ambiguous words (WSJ:NV).

Train	Dev	Type Majority	NN ±3 Words
WSJ:NV	WSJ:NV	93.0	97.0
NV	NV	70.1	77.6

Table 2: Taggers that use no context (Type Majority) or very little context (NN ±3 words) can achieve high accuracies on the ambiguous tokens in the WSJ (WSJ:NV), but would fare much worse on the Noun-Verb dataset.

We therefore aim to create a dataset in which taggers would have to take into account the surrounding context in order to correctly tag ambiguous words, rather than relying on skewed priors per word type. We design a methodology for identifying and labeling hard cases of noun-verb ambiguity. The result is a dataset of over 30,000 hand-labeled, natural, and non-trivial examples of noun-verb ambiguity, which we will make publicly available to facilitate research on modeling for this task.

2.1 Collection Methodology

Our goal is to build a resource which captures a wide range of challenges that a part-of-speech tagger needs to handle in the wild. To produce this resource, we find large sources of *naturally occurring* examples with a diversity of challenges, identify *noun-verb ambiguity*, find the *non-trivial* examples, and finally acquire *high-precision labels* from humans.

2.1.1 Naturally Occurring Sources

All examples come from naturally occurring English web text from three distinct genres. Typical examples from each are shown in Table 3. These genres present a diverse range of challenges: genre 1 has long well-edited sentences, genre 2 makes heavy use of imperative verbs, and genre 3 contains largely headline style short sentences.

2.1.2 Ambiguous Token Detection

We used an online dictionary to identify ambiguous word *types* (such as *play*) that can be either a noun or a verb.[4] To find ambiguous *instances* of these types, we ran a CRF-based tagger similar to Toutanova et al. (2003) over the input sen-

[3]The enhanced tagger that uses both contextual word embeddings and data augmentation (+ELMo+NV Data in Table 1) gets both Example (1) and Example (2) correct.

[4]We exclude a short stop list (*do*, *name*, *state*); the final list contains 24,170 word types.

Representative Examples	Label
Genre 1	
"**Man** With a Vision" peaked at #91 in the UK, spending two weeks on the chart.	**NOUN**
40.7% of the population **benefit** from public assistance as of 2004, up from 23.0% in 2000.	**VERB**
Genre 2	
Your doctor may recommend a diet or **exercise** routine.	**NOUN**
Use within 3 days of cooking.	**VERB**
Genre 3	
Safeguard Infrastructure From Electrical **Surges** & Limit Downtime.	**NOUN**
Stop In Today Or Shop Online!	**VERB**

Table 3: Noun and Verb examples from each genre. All examples are taken from the development set.

tences. We selected tokens tagged as either a noun or a verb[5] and for which the k-best list for that token contained both noun and verb[6] tags with close scores. We used a heuristic that the lower scoring tag had to have a score within 20% of the score of the higher.

2.1.3 Filtering Trivial Examples

Part-of-speech tagging is already a well-established task with plenty of existing labeled examples. Adding more examples similar to *John watched a play* would not affect the output predictions of taggers, which already tend to correctly label tokens as nouns if they follow determiners. Inspired by work on active learning (Tomanek and Hahn, 2009; Small and Roth, 2010), we focused our data collection efforts on difficult examples. To remove easy contexts, we excluded tokens preceded by a determiner or modal verb. Tokens[7] were additionally restricted to be neither adjectival modifiers[8] nor components of noun-compounds[9].

2.1.4 Diversification

Noun-verb disambiguation is a challenge for modern POS taggers both because words can look simultaneously noun- and verb-like to a model, but also because verbs (nouns) can falsely present as nouns (verbs). Our extraction methodology is well-designed to identify the former. To identify tokens on which models are falsely confident, we manually reviewed a sample of tokens discarded in extraction. We found that sentence-initial imperative verbs were very likely to be confidently tagged as nouns. To ensure that this important class of ambiguous tokens was included in our dataset, we made it a special extraction case and did not apply the above filters for trivial examples.

Agreement Type	#	%
Unanimous	23,908	71.4%
Majority	9,122	27.3%
Disagreement	432	1.3%

Table 4: Inter-annotator agreement rates. Unanimous examples had 3/3 agreement, while *majority* examples had 2/3, 3/5, or 4/5 in agreement.

2.1.5 Crowdsourced Annotation

We presented annotators with the extracted tokens in their full sentence context. Annotators were asked to select whether the target word was a "Noun", a "Verb", "Ambiguous", or "Neither" (a noun or a verb). Full annotation guidelines will accompany the dataset release. Each example was annotated by at least three annotators for quality assurance. For batches with larger than average proportions of non-unanimous annotations, the non-unanimous examples were sent to an additional two annotators for a total of five annotations. Table 4 shows that annotators generally had a high level of agreement with each other, with unanimous agreement on 71.4% of the examples and majority agreement on 98.7% of the examples. Annotators achieved an average pace of 40 seconds per sentence.

[5]Nouns and verbs were identified by mapping the fine-grained part-of-speech tag to its coarse-grained category (Petrov et al., 2012): https://github.com/slavpetrov/universal-pos-tags/blob/master/en-ptb.map

[6]We excluded VBN from the set of verb tags, as it often functions more similarly to non-verbs

[7]Specifically, non-sentence initial tokens

[8]Labeled as *amod* according to a dependency parser

[9]Labeled as *nn* according to a dependency parser

Genre	Train	Dev	Test
1	8621	1081	2711
2	6160	919	2289
3	9473	400	1000
All	24254	2400	6000

Table 5: Noun-Verb dataset statistics.

2.2 Final Dataset

To compile the final dataset, we rejected examples in which there was no majority agreement or in which the majority label was "Ambiguous" or "Neither". This excluded 808 sentences and yielded a final dataset size of 32,654. We divided this into training, development, and test sets. Table 5 shows the dataset sizes and genre distributions. The genre distribution of the training set is intentionally different from that of the development and test sets, as realistically one will often have different distributions at training and test time, and future work may want to model this difference (Donmez et al., 2010; Steinhardt and Liang, 2016).

We asked a professional linguist to independently label 200 examples and adjudicate any differences from the crowd-sourced labels with other professional linguists. The linguists found only 7 actual mistakes (3.5% of examples). Of the remaining 96.5% plausible annotations, the linguist agreed with the crowd in 167 cases (83.5%), and found 26 disparities between PTB-style guidelines and plausible intuitive judgments (13%). All but one of the disparities involved a word ending in *"ing"* inside a noun phrase, such as "*Manufacturing* defects"). Also, all but two of the disparities were cases which the crowd source annotators labeled as nouns while the PTB-style guidelines labeled as verbs.

While humans can do well on these instances, Table 2 shows that baseline taggers that use little or no context have high error rates on this dataset, in contrast to the WSJ.

3 Empirical Evaluation of Taggers

In this section, we demonstrate empirically the limitations of several existing taggers on the new challenge dataset. We then take the most accurate, Bohnet et al. (2018), and investigate how it can be enhanced to be much more discriminative in ambiguous contexts. We finish with some error analysis to inspire future work.

3.1 Experimental Setup

Training All experiments used the standard splits of the WSJ Penn Treebank and the new Noun-Verb dataset. Specifically, WSJ Sections 2-21 were used to train all models; where indicated, this was augmented with the training portion of the Noun-Verb dataset. Neural models (Dozat et al. (2017), Bohnet et al. (2018), and extensions) used WSJ Section 22 for early stopping, and were run with $n = 10$ random restarts to compute standard deviations.

Evaluation Models are evaluated on the Noun-Verb test set. The development set was used for developing the proposed enhancements, as well as to do error analysis. To verify performance on the standard task, we also evaluate accuracy on WSJ Section 23, cf. Table 1 first column.

Our evaluation metric is VERB/NON-VERB classification accuracy over tokens which have gold annotations. To evaluate the taggers we map the fine-grained tag output using Petrov et al. (2012): tags with a coarse-grained VERB category map to the VERB label, and all other tags to the NON-VERB label.

3.2 Existing Taggers

We evaluated four commonly used and/or state-of-the-art taggers on our task. The first investigated tagger is the Stanford POS tagger[10] (Toutanova et al., 2003), part of the Stanford CoreNLP Toolkit (Manning et al., 2014) and widely used. This pre-trained model is a log-linear model with features over the surrounding words and tags in a local window around the focus word.

The second investigated tagger is the publicly available *NLP4J*, a pre-trained tagging model (Choi, 2016)[11]. It used feature induction to expand the feature set during training by adding combinations of low-dimensional features. The approach achieved 97.64% on WSJ evaluation. It is worth noting that this model used a large automatically tagged corpus to get ambiguity classes for each word and Choi (2016) showed that this extra piece of information was responsible for the largest part of the improvement.

The third tagger is Dozat et al. (2017), which won the UPOS portion of the CoNLL 2017 Shared Task on Universal Dependencies (Zeman et al.,

[10] https://nlp.stanford.edu/software/tagger.shtml

[11] https://github.com/emorynlp/nlp4j

2017) by a wide margin. It represents each word by a sum of its pretrained word embedding (glove Pennington et al. (2014)), trained word embedding, and the output from an LSTM runs over word's characters. Those representations are supplied to a deep BiLSTM followed by a Multi-Layer Perceptron (MLP) layer. The output from the MLP layer is multiplied by a learned embedding for tags and the tag with the highest score is selected as the output.

Finally the fourth existing tagger is the Meta-BiLSTM (Bohnet et al., 2018) which is the current state of the art on both WSJ and CoNLL 2017 POS tagging evaluation. This model consists of three components, all of which run over the entire input sentence: a word-BiLSTM that takes a sum of pretrained (GloVe (Pennington et al., 2014)) and trained word embeddings, a char-BiLSTM that consumes trained characters embedding and a Meta component that takes a concatenation of word and character representations (at word boundaries) and feeds it to a Bi-LSTM followed by a MLP layer. The final output is computed using softmax over the Meta-MLP representation but a multi-loss is also optimized at the char and word representations level.

For Dozat et al. (2017) and Bohnet et al. (2018), we trained the model on WSJ PTB training data to get comparable models to the two previous systems. For Dozat et al. (2017) we used the default hyperparameters. For Bohnet et al. (2018), the hyperparameters used are almost identical to the original paper.[12]

The first two taggers are linear models (with feature combinations) while the second two are neural models. Both Dozat et al. (2017) and Bohnet et al. (2018) take non-local context into account through BiLSTMs over the full sentence. However, these models might not use this modeling power when trained on the WSJ, since local context is usually sufficient (Table 2).

3.3 Enhancements

We take the best existing tagger (Bohnet et al., 2018) as our starting point to investigate the efficacy of two simple enhancements and their combination for improving noun-verb disambiguation.

The first enhancement is to add generic, contextual word embeddings trained on a billion words of language modeling data (Peters et al., 2018). The second enhancement is to add task-specific targeted training data, with thousands of examples derived from the Noun-Verb training set.

Contextual Word Embeddings (ELMo) The statistics of the new dataset, shown in Table 2, suggest that this dataset might benefit from more contextual modeling. Although the basic Meta-BiLSTM model is already contextual, one can suspect based on the first row in Table 2 that WSJ training might lead the model to ignore wider context. One way to make the model use more contextual information is to replace the word embedding layer with a contextual embedding. We used ELMo embeddings (Peters et al., 2018), which are generated by training a bi-directional language model on a large corpus of unlabeled data. The aim of using ELMo here is that we expect to get different embeddings for a word like "play" when it is used as a verb, as in "I will come and play", versus when it is used as a noun, as in "I liked the two-act play".

We replaced the word embedding layer in the Word component with ELMo.[13] As in Peters et al. (2018), we trained a task specific weighting of the three ELMo layers:

$$v_i^{(\text{word})} = \gamma \sum_{j=0}^{2} s_j \mathbf{h}_{i,j}^{\text{ELMo}}, \quad (1)$$

where $\mathbf{h}_{i,j}^{\text{ELMo}}$ is the j-th layer ELMo embedding of word i, s_j are softmax-normalized weights over the layers, and γ is a scalar parameter. We trained this model on the WSJ training data only.

Targeted Data Augmentation (NV Data) Our Noun-Verb training data comes with gold binary labels ("Noun" or "Verb"). To add them to our current model, we took a simple approach to map the Noun-Verb labels into the fine-grained POS tagset used in the WSJ dataset. To do that, we ran the baseline tagger used to extract the annotated examples in §2.1 over the Noun-Verb training data, and extracted all possible tags for the annotated words, sorted by their score. We then assigned to that word the highest scoring tag consistent with the coarse-grained tags. This resulted in a silver training dataset containing partially labeled sentences, each with one word tagged by its

[12] Two hyperparameter differences: we used two layers instead of three for the word component and a learning rate decay of 0.99994 instead of 0.999994. These were fixed early on and not tuned.

[13] We used the "Original" model from https://allennlp.org/elmo.

Model	SI	¬ SI
Majority class per word type using Noun-Verb training set	74.6	69.3
Existing Taggers		
Toutanova et al. (2003)	47.4	59.6
Choi (2016)	67.8	71.0
Dozat et al. (2017)	68.3	70.7
Bohnet et al. (2018)	68.4±4.0	74.4±0.9
Enhancements		
+ELMo	73.4±2.2	82.1±1.0
+NV Data	**89.3**±0.5	85.4±0.5
+ELMo+NV Data	**90.0**±0.8	**87.6**±0.6

Table 6: Development set accuracies on sentence initial (SI) tokens compared with non-sentence-initial (¬SI) tokens.

fine-grained POS tag. We used this dataset to augment the WSJ training data. Since the Noun-Verb examples only contain one labeled token per sentence, we assigned the unlabeled tokens a cost of zero in the cost function at training time.

ELMo and Data Augmentation Together Finally, we experimented with using both enhancements together. We trained the ELMo-enhanced model on the dataset augmented with the Noun-Verb training set examples. The motivating intuition for combining them is that the inclusion of the difficult Noun-Verb training set examples could encourage the model to make more use of ELMo embeddings than the model trained on the WSJ only. Another possibility is that these two types of enhancements are redundant and that one dominates the other.

3.4 Results

Table 1 shows the main results of both existing taggers and the enhanced models on both WSJ and the Noun-Verb Challenge Set.

Existing Taggers While all four selected taggers achieve accuracies above 97% on WSJ, they all struggle on our noun-verb challenge (Table 1). The widely used tagger of Toutanova et al. (2003) has an accuracy of just 57.6%, below the 70.1% accuracy of a per-word type majority class baseline (Table 2). The best performing tagger (Bohnet et al., 2018) was 3.9% above the next best model. However it still has an error rate of 25%.

The ranking of the four taggers stays the same whether one uses the WSJ or the Noun-Verb Challenge Set for evaluation. However, the magnitude of differences changes drastically. For example, on the WSJ test set, the differences between Dozat et al. (2017) and Toutanova et al. (2003) appear insignificant: Dozat et al. (2017) improves over Toutanova et al. (2003) by 0.09% absolute (3% relative reduction in error). When measured on the Noun-Verb Challenge Set, the differences are stark: the tagger of Dozat et al. (2017) is 12.8% absolute more accurate, which is a 30% relative reduction in error.

Enhancements Experimental results in Table 1 show that ELMo gave 7.2% absolute improvement and did not significantly affect the WSJ results[14]. This is further evidence that WSJ evaluation does not model ambiguities in cases where context matters. Adding the silver Noun-Verb data to the baseline model gave 10% absolute improvement over the baseline. This is significant given that the model capacity remained unchanged. By contrast, hooking up ELMo added a very large multi-layer BiLSTM language model to the parameters.

The best model was the model which used both ELMo embeddings and data augmentation. It achieved 13.1% absolute improvement over the state-of-the-art baseline of Bohnet et al. (2018), equivalent to over a 52% error reduction. This demonstrates that the improvement from ELMo is complementary to that from the additional Noun-Verb data.

Sentence-Initial Examples The trend in Table 1 is magnified in Table 6, which shows development set accuracies separately for tokens that are sentence-initial (SI), which are often imperatives, and for tokens that are not SI.

On SI accuracy, none of the WSJ-trained baselines could beat the most-frequent-tag baseline from the Noun-Verb training data. This shows that these sorts of examples, which are mostly imperatives, are underrepresented in the WSJ corpus. ELMo embeddings were able to improve both SI and non-SI accuracies by roughly the same amount, but again, not as much as adding the Noun-Verb data, which gave a 21.7% boost to SI accuracy. The efficacy of the Noun-Verb data in this case shows that directed training examples can

[14]We also ran the experiment using the "Original (5.5B)" ELMo model, trained on a larger and more diverse corpus. We did not find any significant difference between the two.

Model	Tuning Set WSJ	NV
WSJ Test Set		
Bohnet et al. (2018)	98.00±0.12	97.98±0.13
+ELMo	97.94±0.08	97.85±0.16
+NV Data	97.98±0.11	97.94±0.14
+ELMo+NV Data	97.97±0.09	97.94±0.13
Noun-Verb Test Set		
Bohnet et al. (2018)	74.0±1.2	76.9±0.6 †
+ELMo	82.1±0.9	83.4±0.5 †
+NV Data	86.4±0.4	86.8±0.4
+ELMo+NV Data	88.9±0.3	89.3±0.2 ‡

Table 7: Effect of using different tuning sets. As usual with early stopping, the best tuning set performance was used to evaluate the test set. Here, we evaluated the same experimental runs at two points: when the performance was best on the WSJ development set, and again when the performance was best on the Noun-Verb development set. The increase in Noun-Verb results is significant at the $p < 0.001(\dagger)$ and $p < 0.01(\ddagger)$ levels.

be especially beneficial for fixing some common error patterns.

Impact of Tuning Set Table 7 compares performance of the same experiments on the WSJ and Noun-Verb Challenge test sets, tuned either using the WSJ or the Noun-Verb development set. The only effect of the change in tuning set was for the Noun-Verb tuning to cause the early stopping to sometimes be a little earlier. When we tuned on the Noun-Verb development set, the WSJ results remained almost unchanged, while the Noun-Verb test set results increased significantly. We see that the performance on each dataset is best when matched with its tuning data. The effect was greatest on the unenhanced model, which improved 2.9% absolute on the Noun-Verb evaluation. The best overall Noun-Verb test set result was **89.3**±0.2 when tuned this way.

3.5 Error Analysis

Table 8 shows representative examples that the best baseline run got wrong, along with the predictions from the best runs for each of the different enhancements. While each enhancement reduces all error types, adding Noun-Verb data improves imperatives in particular when compared with adding ELMo. This holds true even when imperatives are not sentence-initial, like the *practice* example in Table 8.

Of the errors made by our best model, roughly a quarter occurred when the focus word was a conjunction. This provides additional evidence for the importance of modeling non-local context in this dataset.

4 Homograph Disambiguation

To show the impact of our best models on a downstream task, we used the text-to-speech homograph disambiguation task described in Gorman et al. (2018). The dataset contains 161 word types, each of which has up to three possible pronunciations. In that work, the authors built a linear model that used lexical features of the focus word and its surrounding words, POS tags, and capitalization, to achieve 95.4% on this task. Here, we want to see the effectiveness of our taggers by using just the POS tag of each word to determine its pronunciation category. To do this, we annotated the homograph disambiguation train and test data with with POS tags using each of our taggers. We collected counts from the training corpus of the form <*word, POS tag, word_sense, Count*>. These counts show how many times a given word got assigned to a certain word sense when it has a certain POS tag. We used those counts to select the most frequent pronunciation for each <*word, POS tag*> pair on the test data. Note that this approach will miss some word senses that cannot be determined from the word and POS tag only, like the difference in pronunciation of the word "jesus" between English: /ˈdʒiːzəs/ and Spanish: /heɪˈsuːs/.

Table 9 shows results for the micro and macro accrucies among different word types in the same way (Gorman et al., 2018) reported their results. The overall results show similar trend to what is observed in the Noun-Verb evaluation results. The Choi (2016), and Bohnet et al. (2018) baseline taggers perform close to the full model in Gorman et al. (2018), which uses a wider context and more features. This is probably due to having a stronger POS tagger than the one used in that model. It is also interesting to see the gap between Toutanova et al. (2003) and the rest of baseline taggers which was measured only on the Noun-Verb evaluation and not in WSJ evaluation. The rest of the results show that using either ELMo achieves a 1.3% absolute improvement over the baseline. while adding data augmentation achieves 0.3% absolute improvement over the baseline. Using both ELMo

Example	Gold	Base	+ELMo	+Data	+ELMo +Data
Will gets his revenge by masquerading as Sue's hairdresser and forcibly shaving her head bald.	**NOUN**	*MD*	**NNP**	*MD*	**NNP**
Will putting a patch over my eye **help** to get the object out of it?	**VERB**	*NN*	**VB**	*NN*	**VB**
If you don't have a table, you can mount the frame on a desk, **stand**, or other structure that will hold the bike off the ground.	**NOUN**	*VB*	*VB*	**NN**	**NN**
For best results, **practice** hitting one note higher than your standard range.	**VERB**	*NN*	*NN*	**VB**	**VB**
Spirit actually suggests unpacking their smokes by rolling the cigarette between your fingers, filter to **end**, so that a pinch or so of tobacco comes out.	**NOUN**	*VB*	*VB*	*VB*	**NN**
Choose the highest combat level and **duel**.	**VERB**	*NN*	*NN*	*NN*	**VB**

Table 8: Development set examples that reflect the types of errors the enhancements address. Base is the tagger of Bohnet et al. (2018), while the remaining columns show the impact of the enhancements. Tags consistent with the gold annotations are in bold and inconsistent are in italics.

Model	Micro	Macro
Best ML system		
Gorman et al. (2018)	95.4	95.1
Existing Taggers		
Toutanova et al. (2003)	91.1	91.5
Choi (2016)	95.8	95.8
Dozat et al. (2017)	94.6	94.7
Bohnet et al. (2018)	95.9±0.2	95.9±0.2
Enhancements		
+ELMo	**96.7**±0.2	**96.7**±0.2
+NV Data	96.2±0.2	96.2±0.2
+ELMo+NV Data	**96.7**±0.3	**96.7**±0.3

Table 9: Accurcies of different models on the homograph disambiguation test set. All enhancements' improvements over (Bohnet et al., 2018) baseline are statistically significant $p < 0.008$. Standard deviations are estimated from $n = 10$ random restarts, and p-values were computed using a heteroscedastic two-tailed t-test.

and data augmentation was not better than just using ELMo. Those improvements correspond to a 28% error reduction compared to the machine-learned model in Gorman et al. (2018).

5 Discussion and Related Work

Dataset Creation Prior work in crowd-sourcing syntactic annotations and using them in models motivated the dataset creation portion of this work. Jha et al. (2010) showed that non-linguists could reliably do aspects of syntactic annotation, and Hovy et al. (2014) showed that non-experts could annotate universal part-of-speech tags (Petrov et al., 2012) almost as well as experts. He et al. (2016) then showed that incorporating crowd-sourced annotations improves parsing by a noticeable margin on the *subset* of sentences in which the human judgments affected the parser's output. Inspired by this result, we focused our efforts on collecting annotations that were likely to change a tagger's predictions and humans can annotate reliably.

This work filtered out trivial examples via hand-written heuristics targeted towards examples that taggers generally get correct (Section 2.1). One interesting direction for future work would be to eliminate this manual step. One option could be to instead use automatically produced high-precision interpretable rules to filter out these examples, such as the Anchor explanations output by Ribeiro et al. (2018). Table 1 in that paper shows how the system can automatically induce that a part-of-speech tagging system will tag the word *play* as a NOUN in the sentence *I went to a play yesterday* because the previous word is a determiner.

Measurement Manning (2011) performed an error analysis for WSJ and discovered that 19% of the errors fall under "Difficult linguistics" which need non-local context modeling to be able to

solve them. The negative results of Kiddon et al. (2015) on using existing supervised part-of-speech taggers for imperative detection provided motivation for focusing on noun-verb confusion. However we are not aware of any prior work on trying to measure part-of-speech-tagging accuracy on hard ambiguities that are easily recognized by human using diverse corpora.

6 Conclusion and Future Work

This paper proposes a challenge set approach to evaluating part-of-speech taggers, and builds a new resource for doing so. We show that a part-of-speech tagger can be trained to be better at noun-verb ambiguity by using extra Noun-Verb targeted training data or by adding contextual word embedding. We also show that our evaluation data can measure improvements in Noun-Verb disambiguation that standard evaluation dataset was not able to capture. Those previously unmeasured improvements in the Noun-Verb disambiguation are shown to lead to improvements in a downstream task. Improvements were especially large on sentence-initial tokens, which are often imperatives. Even with these improvements, there is still a large gap between the noun-verb accuracies and overall WSJ tagging accuracy. We expect that closing this gap will make incorporating syntax more useful across natural language understanding applications.

Future work can include exploring ways to incorporate more context into the tagger, possibly by using information from dependency tree. Also investigating more downstream tasks and explore if this dataset can be used directly in downstream tasks in a way similar to what have been done in Swayamdipta et al. (2017) and (Eriguchi et al., 2017; Niehues and Cho, 2017; Kiperwasser and Ballesteros, 2018) for injecting syntax in semantic role labeling and translation tasks. A third direction for research would be using this dataset to evaluate different contextual modeling approaches and investigate the creation and using such context sensitive dataset to create simpler and smaller models that can capture a lot of contextual word representation.

Future work on dataset creation can include generating similar challenge datasets for different key ambiguities in NLP. A collection of such datasets could be one way to cover hard examples that models do not get right but humans are good at. Such targeted datasets can complement the use of large unsupervised contextual embedding models. This can open an avenue to improve core NLP tasks on hard relevant ambiguities that allows making progress on downstream tasks.

Acknowledgments

The authors gratefully acknowledge their anonymous reviewers for their insightful questions and feedback. We are grateful to Alexander Clines, Kazoo Sone, Ashwin Kakarla, Daphne Luong, and Austin Tarango for their assistance in the creation of the dataset. Thanks also go to members of the Google AI Language group for their input to this work, including Slav Petrov, Michael Collins, and all who provided feedback in reading group sessions.

References

Bernd Bohnet, Ryan McDonald, Gon¸calo Simões, Daniel Andor, Emily Pitler, and Joshua Maynez. 2018. Morphosyntactic tagging with a meta-bilstm model over context sensitive token encodings. In *(To Appear) Proceedings of ACL*.

Jinho D. Choi. 2016. Dynamic feature induction: The last gist to the state-of-the-art. In *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 271–281, San Diego, California. Association for Computational Linguistics.

Michael Collins. 2002. Discriminative training methods for hidden markov models: Theory and experiments with perceptron algorithms. In *Proceedings of the ACL-02 conference on Empirical methods in natural language processing-Volume 10*, pages 1–8. Association for Computational Linguistics.

Ronan Collobert, Jason Weston, Léon Bottou, Michael Karlen, Koray Kavukcuoglu, and Pavel Kuksa. 2011. Natural language processing (almost) from scratch. *Journal of Machine Learning Research*, 12(Aug):2493–2537.

Pinar Donmez, Guy Lebanon, and Krishnakumar Balasubramanian. 2010. Unsupervised supervised learning i: Estimating classification and regression errors without labels. *Journal of Machine Learning Research*, 11(Apr):1323–1351.

Timothy Dozat, Peng Qi, and Christopher D. Manning. 2017. Stanford's graph-based neural dependency parser at the conll 2017 shared task. In *Proceedings of the CoNLL 2017 Shared Task: Multilingual Parsing from Raw Text to Universal Dependencies*, pages 20–30, Vancouver, Canada. Association for Computational Linguistics.

Akiko Eriguchi, Yoshimasa Tsuruoka, and Kyunghyun Cho. 2017. Learning to parse and translate improves neural machine translation. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 72–78, Vancouver, Canada. Association for Computational Linguistics.

Kyle Gorman, Gleb Mazovetskiy, and Vitaly Nikolaev. 2018. Improving homograph disambiguation with supervised machine learning. In *Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC 2018)*.

Luheng He, Julian Michael, Mike Lewis, and Luke Zettlemoyer. 2016. Human-in-the-loop parsing. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 2337–2342.

Dirk Hovy, Barbara Plank, and Anders Søgaard. 2014. Experiments with crowdsourced re-annotation of a pos tagging data set. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (volume 2: Short Papers)*, volume 2, pages 377–382.

Mukund Jha, Jacob Andreas, Kapil Thadani, Sara Rosenthal, and Kathleen McKeown. 2010. Corpus creation for new genres: A crowdsourced approach to pp attachment. In *Proceedings of the NAACL HLT 2010 Workshop on Creating Speech and Language Data with Amazon's Mechanical Turk*, pages 13–20. Association for Computational Linguistics.

Chloé Kiddon, Ganesa Thandavam Ponnuraj, Luke Zettlemoyer, and Yejin Choi. 2015. Mise en place: Unsupervised interpretation of instructional recipes. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 982–992.

Eliyahu Kiperwasser and Miguel Ballesteros. 2018. Scheduled multi-task learning: From syntax to translation. In *Transactions of the Association for Computational Linguistics*, volume 6, page 225–240.

Christopher Manning, Mihai Surdeanu, John Bauer, Jenny Finkel, Steven Bethard, and David McClosky. 2014. The stanford corenlp natural language processing toolkit. In *Proceedings of 52nd annual meeting of the association for computational linguistics: system demonstrations*, pages 55–60.

Christopher D Manning. 2011. Part-of-speech tagging from 97% to 100%: is it time for some linguistics? In *International Conference on Intelligent Text Processing and Computational Linguistics*, pages 171–189. Springer.

Jan Niehues and Eunah Cho. 2017. Exploiting linguistic resources for neural machine translation using multi-task learning. *WMT 2017*, pages 80–89.

Jeffrey Pennington, Richard Socher, and Christopher Manning. 2014. Glove: Global vectors for word representation. In *Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP)*, pages 1532–1543.

Matthew E Peters, Mark Neumann, Mohit Iyyer, Matt Gardner, Christopher Clark, Kenton Lee, and Luke Zettlemoyer. 2018. Deep contextualized word representations. *arXiv preprint arXiv:1802.05365*.

Slav Petrov, Dipanjan Das, and Ryan McDonald. 2012. A universal part-of-speech tagset. In *Proceedings of the Eighth International Conference on Language Resources and Evaluation (LREC-2012)*.

Marco Tulio Ribeiro, Sameer Singh, and Carlos Guestrin. 2018. Anchors: High-precision model-agnostic explanations. In *Proceedings of AAAI*.

Rico Sennrich and Barry Haddow. 2016. Linguistic input features improve neural machine translation. In *Proceedings of the First Conference on Machine Translation*, pages 83–91, Berlin, Germany. Association for Computational Linguistics.

Kevin Small and Dan Roth. 2010. Margin-based active learning for structured predictions. *International Journal of Machine Learning and Cybernetics*, 1(1-4):3–25.

Richard Sproat, Julia Hirschberg, and David Yarowsky. 1992. A corpus-based synthesizer. In *Second International Conference on Spoken Language Processing*.

Jacob Steinhardt and Percy S Liang. 2016. Unsupervised risk estimation using only conditional independence structure. In *Advances in Neural Information Processing Systems*, pages 3657–3665.

Swabha Swayamdipta, Sam Thomson, Chris Dyer, and Noah A Smith. 2017. Frame-semantic parsing with softmax-margin segmental rnns and a syntactic scaffold. *arXiv preprint arXiv:1706.09528*.

Katrin Tomanek and Udo Hahn. 2009. Semi-supervised active learning for sequence labeling. 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 1039–1047. Association for Computational Linguistics.

Kristina Toutanova, Dan Klein, Christopher D Manning, and Yoram Singer. 2003. 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.

Daniel Zeman, Martin Popel, Milan Straka, Jan Hajic, Joakim Nivre, Filip Ginter, Juhani Luotolahti,

Sampo Pyysalo, Slav Petrov, Martin Potthast, Francis Tyers, Elena Badmaeva, Memduh Gokirmak, Anna Nedoluzhko, Silvie Cinkova, Jan Hajic jr., Jaroslava Hlavacova, Václava Kettnerová, Zdenka Uresova, Jenna Kanerva, Stina Ojala, Anna Missilä, Christopher D. Manning, Sebastian Schuster, Siva Reddy, Dima Taji, Nizar Habash, Herman Leung, Marie-Catherine de Marneffe, Manuela Sanguinetti, Maria Simi, Hiroshi Kanayama, Valeria dePaiva, Kira Droganova, Héctor Martínez Alonso, ÇağrÄ Çöltekin, Umut Sulubacak, Hans Uszkoreit, Vivien Macketanz, Aljoscha Burchardt, Kim Harris, Katrin Marheinecke, Georg Rehm, Tolga Kayadelen, Mohammed Attia, Ali Elkahky, Zhuoran Yu, Emily Pitler, Saran Lertpradit, Michael Mandl, Jesse Kirchner, Hector Fernandez Alcalde, Jana Strnadová, Esha Banerjee, Ruli Manurung, Antonio Stella, Atsuko Shimada, Sookyoung Kwak, Gustavo Mendonca, Tatiana Lando, Rattima Nitisaroj, and Josie Li. 2017. Conll 2017 shared task: Multilingual parsing from raw text to universal dependencies. In *Proceedings of the CoNLL 2017 Shared Task: Multilingual Parsing from Raw Text to Universal Dependencies*, pages 1–19, Vancouver, Canada. Association for Computational Linguistics.