# Weakly Supervised Headline Dependency Parsing

**Adrian Benton**[*] Bloomberg
**Tianze Shi**[*] Cornell University
{adbenton,tianze}@google.com

**Ozan İrsoy** Bloomberg
**Igor Malioutov** Bloomberg
{oirsoy,imalioutov}@bloomberg.net

## Abstract

English news headlines form a register with unique syntactic properties that have been documented in linguistics literature since the 1930s. However, headlines have received surprisingly little attention from the NLP syntactic parsing community. We aim to bridge this gap by providing the first news headline corpus of Universal Dependencies annotated syntactic dependency trees, which enables us to evaluate existing state-of-the-art dependency parsers on news headlines. To improve English news headline parsing accuracies, we develop a projection method to bootstrap silver training data from unlabeled news headline-article lead sentence pairs. Models trained on silver headline parses demonstrate significant improvements in performance over models trained solely on gold-annotated long-form texts. Ultimately, we find that, although projected silver training data improves parser performance across different news outlets, the improvement is moderated by constructions idiosyncratic to outlet.

## 1 Introduction

English news headlines are written to convey the most salient piece of information in an article in as little space as possible. This makes them an attractive target for information extraction systems, and other NLP applications that operate on the most salient information in a news article. Headlines have been the target for many NLP tasks including semantic clustering (Wities et al., 2017; Laban et al., 2021), multi-document summarization (Bambrick et al., 2020), and sentiment/stance classification (Strapparava and Mihalcea, 2007; Ferreira and Vlachos, 2016).

However, while headlines often present the most salient information from an article, brevity introduces its own obstacles. English news headlines are written in a unique register known as *headlinese*. The structure of this register is determined primarily by typographical constraints along with the various functions that headlines serve, including summarization and eliciting reader interest (Mårdh, 1980). Headlinese syntax deviates from long-form news body through such features as a preference for atypical word senses and terms, frequent omission of determiners and auxiliaries, the acceptability of nominal and adverbial phrases, as well as multiple independent phrases in a single headline (decks). Figure 1 presents a sample of English headlines exhibiting some of these properties, and parse errors made by a strong English dependency parser, Stanza (Qi et al., 2020).

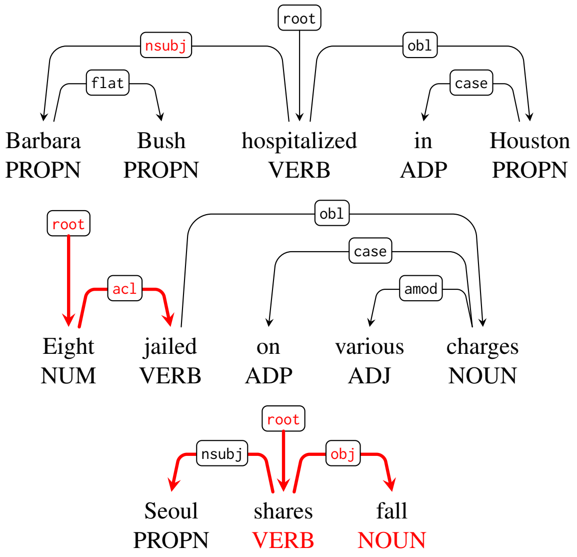


Figure 1: Example English news headlines with parses and POS tags generated by Stanza (Qi et al., 2020). Mispredicted relations and labels in red. The text shown is after truecasing, before being fed to Stanza.

While the syntax of English headlines deviates significantly from article body text, there has been little work in evaluating and developing classical NLP pipeline models for this register. News headline-related NLP tasks, such as headline gener-

[*] Now at Google Research. Work done while at Bloomberg and Cornell University, respectively.

*Findings of the Association for Computational Linguistics: EMNLP 2022*, pages 6549 - 6564
December 7-11, 2022 ©2022 Association for Computational Linguistics

ation (Rush et al., 2015; Takase et al., 2016; Takase and Okazaki, 2019) and classification (Kozareva et al., 2007; Oberländer et al., 2020), do not rely on syntactic annotations like POS tags, syntactic or semantic parses. This is by design, as the oftentimes poor performance of existing syntactic parsers on headlinese has impeded their application to tasks such as sentence compression (Filippova and Altun, 2013).

In this work, we take a step towards improving headline dependency parsers by releasing the first English news headline treebank annotated according to universal dependency (UD) typology. We present the first quantitative evaluation of existing dependency parsers on English headlinese, and we propose a method for generating weak supervision for headline dependency parsers inspired by cross-lingual annotation projection (Yarowsky et al., 2001).

**Contributions**

1. We release the first English headline treebank of 1,055 manually annotated and adjudicated universal dependency (UD) syntactic dependency trees, the **E**nglish **H**eadline **T**reebank (EHT), to encourage research in improving NLP pipelines for English headlinese.[1]

2. We establish baselines on the EHT evaluation set with existing state-of-the-art parsers. Our experiments confirm prior observations that existing syntactic parsers perform poorly on headlinese (Filippova and Altun, 2013).

3. We propose a tree projection method to generate weak supervision for training more accurate headline parsers, and demonstrate that training on silver-annotated trees can significantly reduce parsing errors. Most strikingly, we show that that after finetuning on weak supervision, we are able to reduce root prediction relative error rate by 92.8% within domain, and by 21.3% for an out-of-domain wire. We further show that these gains translate to downstream improvements in the quality of tuples extracted by an open domain information extraction system.

This paper is structured as follows: Section 2 presents prior work on linguistic analyses of headlinese and their treatment in the NLP community; Section 3 describes the EHT annotation process and descriptive statistics; Section 4 describes our tree projection algorithm for generating silver headline dependency trees; Section 5 describes our experiment set up; Sections 6 and 7 respectively present intrinsic parser performance on the EHT and extrinsic performance on an open information extraction (OpenIE) task; and Section 8 presents related work on headline and low-resource syntactic processing.

## 2 Background

**Linguistic Analysis of Headlines** English news headlines are known for their compressed telegraphic style, constituting a unique register known as *headlinese* (Garst and Bernstein, 1933; Straumann, 1935). Through a manual corpus analysis of over 1,800 headlines from two British newspapers, Mårdh (1980) finds that headlinese shares some syntactic features with "ordinary" English language, but there also exist a number of features peculiar to headlinese. These include the validity of nominal and adverbial headlines, lack of determiners, omission of auxiliaries and copulas, and use of the present tense to denote urgency of the event. Nevertheless, these syntactic hallmarks of headlinese vary across country of publication (Ehineni, 2014), news outlet (Mårdh, 1980; Siegal and Connolly, 1999), and time period (Vanderbergen, 1981; Schneider, 2000; Afful, 2014), making development of a strong, general English headline parser particularly challenging.

**Headline NLP** In spite of the clear evidence that headlinese differs significantly from standard written English syntax, there has been scant work on building traditional NLP pipeline components for headlines. This has limited the linguistic features that NLP researchers can extract from headlines, and subsequently limited the analyses that can be performed on them. For instance, Filippova and Altun (2013) reports that poor parsing accuracy for headlines impedes their use of headline parses in alignment with a body sentence.

This is not to say that headlines have been ignored as an object of study by the community. Tasks such as headline generation, compression, and news summarization are all well-studied problems, partly because they circumvent the need for annotation of linguistic structure (Filippova and Altun, 2013; Rush et al., 2015; Tan et al., 2017; Takase and Okazaki, 2019; Ao et al., 2021). Other studied tasks include emotion identification/senti-

[1] The EHT, licensed under CC-4.0, is available at https://github.com/bloomberg/emnlp22_eht

|  | Dataset | Headlines | Tokens |
|---|---|---|---|
| EHT | GSC | 600 | 5,017 |
|  | NYT | 455 | 3,986 |
| Silver | Projection | 48,633 | 395,237 |

Table 1: Statistics of our gold (EHT) data and silver (projected GSC) data.

ment analysis (Kozareva et al., 2007; Oberländer et al., 2020), stance identification (Ferreira and Vlachos, 2016), framing or bias detection (Gangula et al., 2019; Liu et al., 2019), and headline clustering (Laban et al., 2021).

## 3 The English Headline Treebank

Here we describe the compilation and characteristics of our evaluation set, the **E**nglish **H**eadline **T**reebank (EHT).

### 3.1 Data Sources and Pre-processing

We sample English news headlines from two sources to build the EHT: the Google sentence compression corpus (GSC; Filippova and Altun, 2013), and the New York Times Annotated Corpus[2] (NYT). We sample from the GSC as it contains hundreds of thousands of news headlines across tens of thousands of domains, and as is described in Section 4, we leverage it as a rich source of silver-annotated training data. We sample 600 headlines from GSC in total.

In addition, we sample from NYT to form an out-of-domain evaluation set, which was not subjected to the same preprocessing decisions used to build the GSC. We sample 500 headlines uniformly at random from the NYT, under the constraint that they are 4 to 12 tokens long (up to the 95th percentile). We impose this length constraint to avoid trivial parses, as well as noise in the data.[3] Of these 500 headlines, we removed 45 headlines that were templated death notices and obituaries.[4]

All headlines are tokenized using the Stanford Penn Treebank tokenization algorithm with default settings. We use Stanza (Qi et al., 2020) to bootstrap our expert annotators with predicted UD-style part-of-speech tags and parse trees. To reduce the discrepancy between training data of Stanza and our news headline data, we truecased headlines using an n-gram truecaser model trained on English news body text, and inserted a period at the end of the headline before running inference with Stanza. Table 1 shows the number of headlines and tokens in our headline datasets.

### 3.2 Annotation and Adjudication

Our annotation follows the UD guidelines whenever possible. In Appendix A, we provide an addendum for consistent treatment of syntactic constructions that frequently occur in headlines, but are underspecified in the original UD guidelines.

In the first stage of annotation, each headline is independently annotated for POS tag sequence and dependency parse by two expert annotators[5] using the UD Annotatrix interface (Tyers et al., 2017). Any discrepancies between annotators are resolved in the second, adjudication, stage. Two adjudicators independently examine the annotations and pick the one that conforms to UD guidelines, or construct their own parse if they disagree with both candidate parses from the first stage.[6] The third and last stage is group discussion, where all four annotators and adjudicators discuss and resolve any remaining disagreements.

First-stage annotation takes roughly one minute per instance per annotator, and the combination of second and third-stage adjudication takes another minute per instance per adjudicator. On a sample of 50 headlines held out to compute inter-annotator agreement, we find that 56% of headlines were parsed or POS-tagged incorrectly by Stanza. 48% contain some attachment error and 50% have an incorrect relation label. Annotators achieved a 72% headline-level agreement rate on this sample in the first stage, with individual POS tag agreement of 98.8% and labeled dependency attachment agreement rate of 94.8%. Many annotator discrepancies arose from parsing the internal structure of named entities, which were resolved during the subsequent adjudication phases. These along with other common issues are listed in Appendix A.

### 3.3 Characterizing Headline Data

Figure 2 presents the distributions of relation labels in the EHT compared to the UD 2.8 English

[2] https://catalog.ldc.upenn.edu/LDC2008T19

[3] For examples, occasionally the NYT headline field appeared to contain a full article body.

[4] e.g., ”Paid Notice : Deaths GOLDBERG , HERBERT”.

[5] All four annotators and adjudicators are fluent English speakers, have a background in NLP, and were trained in the UD guidelines before annotation.

[6] During adjudication, the identities of the first-stage annotators are anonymized to avoid biasing towards or against any particular annotator.

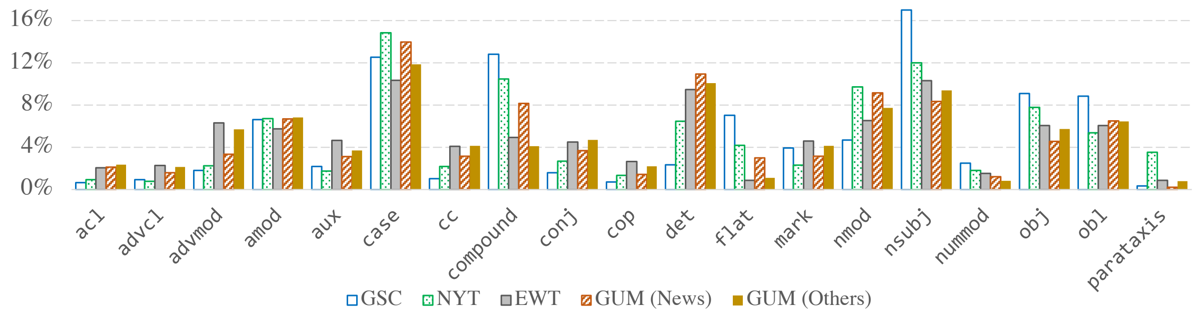


Figure 2: Distributions of dependency relation labels across the EHT, compared with UD 2.8 EWT and GUM corpora. We exclude `punct` and `root` relations when calculating the distributions, and omit low-frequency labels (below 2% across all datasets) in this chart.

Web Treebank (EWT; Silveira et al., 2014) and the Georgetown University Multilayer corpus (GUM; Zeldes, 2017). The EWT includes data from web media (weblogs, newsgroups, emails, reviews, and Yahoo! answers), and the texts in GUM corpus are drawn from a range of domains including news, fiction, academic writings, as well as dialogue such as transcribed interviews.

Compared with texts from other domains, English news headlines use fewer determiners, auxiliaries, and copulas, which is consistent with prior linguistic characterization of headlinese (Mårdh, 1980). News headlines have higher proportions of `compound` and `flat` relations, due to frequent mentions of named entities, and we also observe larger percentages of `nsubj` and `obj` relations, as a consequence of headline brevity and focus on core argument structure.

**Algorithm 1** Algorithm for projecting a parse tree from a news article lead sentence $\mathsf{s}$ to a headline $\mathsf{h}$, which is a subsequence of $\mathsf{s}$.

```
Definitions
$N(s)$ : set of nodes in tree $s$, each corresponding to a word in the sentence or the dummy root
$N(h) \subseteq N(s)$ : set of nodes in tree $h$

function EXTRACTSUBTREE($s, h$)
    $N' \leftarrow \varnothing$        ▷ Subset of nodes to be returned
    for all $n \in N(h)$ do
        $P \leftarrow$ nodes on path from root of $s$ to $n$
        $N' \leftarrow N' \cup P$
    end for
    $R' \leftarrow$ relations in tree $s$ s.t. both the head and tail of the relation are in $N'$
    return $N', R'$
end function

function PROJECT($s, h$)
    $N', R' \leftarrow$ EXTRACTSUBTREE($s, h$)
    while $|N'| > |N(h)|$ do
        $n \leftarrow$ closest to root s.t. $n \in N', n \notin N(h)$
        $c \leftarrow$ leftmost child of $n$ s.t. $c \in N'$
        $p \leftarrow$ parent of $n$ according to $R'$
        Update $R'$: attach all siblings of $c$ to $c$ and attach $c$ to $p$
        $N' \leftarrow N' \setminus \{n\}$
    end while
    return A tree formed by $N', R'$
end function
```

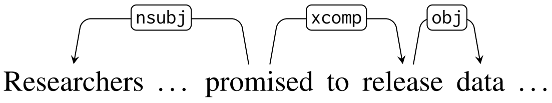
## 4 Generating Silver Data by Projecting from Lead Sentences

While EHT is suitable for evaluating parser performance on English news headlines, 1,055 headlines is much less data than is typically used for training a syntactic parser. For comparison, the EWT contains more than 15 times as many tokens as the EHT.

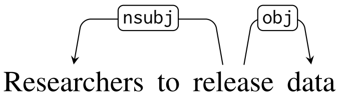
On the other hand, it may be data-inefficient to manually annotate a training set of tens of thousands of headlines, since English news headlines constitute a different register of written English, not a different language. Although certain constructions are idiosyncratic to headlines, one can often expand a headline to a well-formed sentence in the news body register, as words are frequently omitted to produce a headline (Straumann, 1935; Mårdh, 1980). This section describes an algorithm for automatically assigning dependency trees to unannotated headlines to create silver training data for training a headline dependency parser.

Our approach is based on the key observation that headlines convey similar semantic content as the bodies and they typically share many local substructures. Lead sentences, often the first sentence in an article, serve a similar function as news headlines in grabbing reader attention and stating essential facts about news events; lead sentences are sometimes direct expansions of the headlines. Consequently, the pairs of lead sentence and headline have been used to automatically construct examples for sentence compression (Filippova and Altun, 2013).

Algorithm 1 projects the dependency tree annotation from a news article lead sentence to a headline, where the headline is a (possibly non-contiguous) subsequence of the lead sentence. The main idea of this algorithm is to prune the lead sentence's dependency tree until it only contains those tokens in the headline. When a token from the lead sentence is missing in the headline, but it has children appearing in both strings, we promote its first child to preserve connectivity. For example, the following sentence snippet contains an extra "promised" than the corresponding headline:



and our algorithm promotes "release" to be the new root of the tree for the headline:



We use Algorithm 1 to construct a silver-annotated corpus of headline dependency trees from headline-lead sentence pairs in the GSC corpus. Our silver corpus contains 48,633 headlines that satisfied the subsequence constraint, the same magnitude as the EWT and significantly larger than our manually-annotated EHT. Of these, 8,633 were held out as a development set, with the remaining 40,000 used for training.

## 5 Training Headline Parsers

We vary two main dimensions during parser training: training data selection and data combination methods. We consider the EWT, the projected GSC headline data described in Section 3, and a combination of both as different training sets. We experiment with three different ways of combining training sets: a) simply concatenate the two corpora; b) use a multi-domain[7] model with shared feature extractors, but independent parameters for the parsing modules in each domain (Benton et al., 2021); and c) first train on the gold-standard EWT corpus, and subsequently finetune on the silver-annotated GSC headline corpus.

**Model** Our model architecture follows the deep biaffine parser (Dozat and Manning, 2017), using a pre-trained BERT (Devlin et al., 2019) as a feature extractor. This architecture underlies many state-of-the-art dependency parsers (e.g., Kondratyuk and Straka, 2019) and the winning solutions in recent runs of IWPT shared tasks (Bouma et al., 2020, 2021). Model and implementation details are provided in Appendix C.

**Combining EWT and Projected GSC** In our experiments, we consider three different data combination methods:

1. **Concat**: We simply concatenate the two corpora and train a dependency parser based on the joint dataset. This strategy does not require any modification to the model architecture or the training procedures.

2. **MultiDom**: Inspired by the multi-domain POS tagging architecture in Benton et al. (2021), we experiment with a multi-domain parser. In this parsing architecture, we have one parser for EWT and another for headlines, sharing the same underlying BERT-based feature extractor. In other words, each parser has its own trainable projection and biaffine attention layers. In each training step, we sample a batch of examples from the concatenated corpora and jointly update the domain-specific parameters and shared feature extractor.

3. **Finetune**: Finally, we also experiment with a two-step training strategy where we finetune on the projected GSC headline data based on a trained parser on EWT. Stymne et al. (2018) find this strategy to be one of the most effective ways to learn from multiple treebanks in the same language.[8]

[7] We abuse terminology here and use *domain* to refer to examples that come from different corpora, even if the distinction between language in each corpus is the register.

[8] They report another strategy of using treebank embeddings to be equally effective as finetuning, but that requires modification to the model by adding treebank embeddings and can be viewed as a simplification to our multi-domain parser.

**Ensembling** We train each parser under each setting with five random restarts and report means and standard deviations in Section 6. To reduce variations in our manual analysis in Section 7, we analyze the ensembled parse trees using the reparsing technique of Sagae and Lavie (2006).

## 6 Intrinsic Parser Performance

Intrinsic parser performance is shown in Table 2. The discrepancy in baseline performance between NYT and GSC (85.49% vs. 60.60% LAS) can be attributed to the fact that NYT headlines exhibit a closer distribution of relation types to EWT than GSC headlines (Figure 2). Many NYT headlines already constitute a well-formed body sentence, albeit without final punctuation. This is further supported by the fact that only training on projected GSC parse trees significantly improves performance on GSC (89.09% LAS) while actually hurting NYT performance, with a slight drop to 84.75% LAS.

However, training on both EWT and projected GSC improves parser performance across both domains. We found that training a multi-domain model performed about as well as concatenating EWT and projected GSC training data. Ultimately, we found that a pipelined finetuning scheme – first training on EWT, then silver projected GSC headlines – yielded a strong parser across both domains (LAS of 87.13% on NYT and 90.08% on GSC).

### 6.1 Error Analysis

Although finetuning on GSC headlines with projected dependency parses improves parser performance on both the GSC and NYT evaluation sets, we see more marked improvements on the GSC corpus. Figure 3 displays the % relative error reduction in F1 of the GSC-finetuned ensemble against the EWT baseline ensemble broken by relation type. See Appendix D for absolute F1 for each relation type and domain. We compute model performance for all models using the `eval07.pl` evaluation script released as part of the First Shared Task on Parsing Morphologically-Rich Languages.[9]

It is clear from the relation-level error analysis that most of the gains on GSC come from correct identification of the headline root, arguably the most important relation in the headline parse. In fact, the finetuned parser achieves 98.2% recall in identifying the root, whereas the baseline parser only achieves 74.6% recall. Headlines using the "to VERB" construction, indicating future tense or an expected event, are particularly susceptible to root misprediction by the baseline parser (example given in Figure 4). Performance on the `nsubj` relation also improves as a side effect of correctly identifying the root.

Gains on the NYT evaluation set are consistent across most relations, but with smaller improvements. This is encouraging in that no NYT headline training data was used to train the model, silver or otherwise. `parataxis` benefits from finetuning on projected GSC headlines. This relation occurs frequently in NYT headlines due to a preference for headlines with multiple decks, independent syntactic components: e.g., "*Essay ; B.C.C.I. : Justice Delayed*". The fact that this deck structure occurs more frequently in headlines results in a parser with a stronger prior for predicting `parataxis`.

It is also important to note that the finetuned parser can identify passive constructions much more accurately than the baseline. % F1 performance for identifying `nsubj:pass` improves from 11.1% to 90.5% on GSC and from 60.0% to 86.8% on NYT.

## 7 Extrinsic Evaluation

In addition to intrinsic evaluation of parsers, we also evaluate these models downstream. We perform an extrinsic evaluation using the state-of-the-art syntax-based PredPatt OpenIE System (White et al., 2016), and evaluate extracted tuples using the protocol and error typology taken from Benton et al. (2021). As PredPatt relies solely on a UD parse and POS tag sequence to extract candidate tuples, this constitutes a direct downstream evaluation of more accurate headline parses.

**OpenIE Evaluation Protocol** Two annotators independently annotated 200 extracted tuples manually.[10] These tuples were randomly sampled from the GSC and NYT headlines, such that the PredPatt extracted different OpenIE tuples from the baseline ensemble parse compared to finetuned ensemble parse. Each tuple was judged as either *Correct*, or annotated with its most salient error type: *Malformed Predicate*, *Bad Sub-predicate*, *Missing Core Argument*, *Argument Misattachment*, or *Incomplete Argument*.

[9] `http://www.spmrl.org/spmrl2014-sharedtask.html`

[10] A subset of the annotators from the dependency parse annotations.

| Training data / regime | NYT | | | | GSC | | | |
|---|---|---|---|---|---|---|---|---|
| | UAS | LAS | UEM | LEM | UAS | LAS | UEM | LEM |
| EWT | 88.82$_{\pm 0.22}$ | 85.49$_{\pm 0.28}$ | 57.89$_{\pm 0.65}$ | 48.57$_{\pm 1.03}$ | 83.01$_{\pm 0.36}$ | 80.60$_{\pm 0.28}$ | 50.80$_{\pm 1.14}$ | 42.90$_{\pm 0.68}$ |
| Proj | 88.27$_{\pm 0.30}$ | 84.75$_{\pm 0.33}$ | 56.88$_{\pm 1.17}$ | 48.22$_{\pm 0.63}$ | 90.99$_{\pm 0.23}$ | 89.09$_{\pm 0.30}$ | 68.33$_{\pm 0.59}$ | 59.93$_{\pm 0.80}$ |
| Concat | **89.97**$_{\pm 0.65}$ | 87.05$_{\pm 0.56}$ | 60.70$_{\pm 1.35}$ | 53.14$_{\pm 1.66}$ | 91.23$_{\pm 0.19}$ | 89.32$_{\pm 0.23}$ | 68.67$_{\pm 0.67}$ | 61.23$_{\pm 0.83}$ |
| MultiDom | 89.58$_{\pm 0.27}$ | 86.29$_{\pm 0.36}$ | 60.00$_{\pm 0.87}$ | 50.29$_{\pm 0.90}$ | 91.16$_{\pm 0.11}$ | 89.31$_{\pm 0.21}$ | 68.63$_{\pm 0.77}$ | 61.07$_{\pm 1.00}$ |
| Finetune | 89.93$_{\pm 0.14}$ | **87.13**$_{\pm 0.18}$ | **61.45**$_{\pm 0.65}$ | **53.71**$_{\pm 0.75}$ | **91.88**$_{\pm 0.06}$ | **90.08**$_{\pm 0.11}$ | **71.07**$_{\pm 0.25}$ | **63.37**$_{\pm 0.27}$ |

Table 2: Parsing accuracies on NYT and GSC headlines from the EHT, comparing models trained on EWT, silver headline projection data (Proj), and different methods for combining these two training data sources: concatenating (Concat), training with a multi-domain model (MultiDom), and finetuning on silver GSC headline trees (Finetune). UAS and LAS correspond to (un)labeled attachment score, and UEM/LEM to (un)labeled exact match score (at the sentence level).

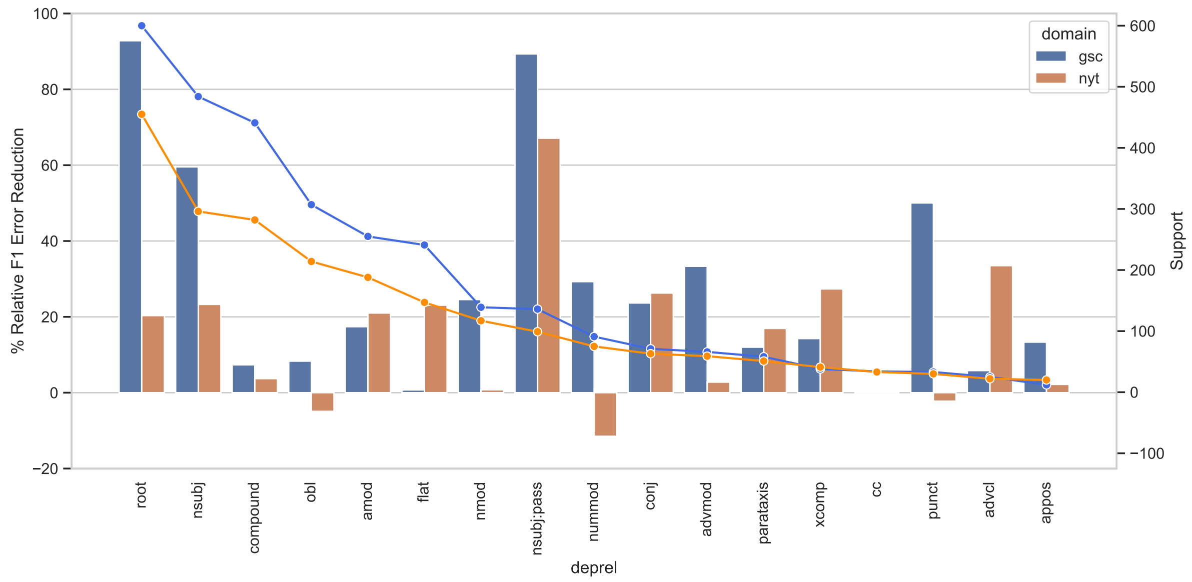


Figure 3: % relative error reduction in F1 score across dependency relations for both the GSC and NYT evaluation sets, from the ensembled *Both (finetuning)* model to *EWT (baseline)*. Relations are sorted by descending frequency and only relations that occurred at least 20 times in the evaluation set are shown. Support for each class is indicated by the line plot.

To control for potential annotation bias, tuples were shuffled and identity of the parser and example domain were hidden from annotators. After independent annotation, the two annotators adjudicated conflicting annotations and converged on a single label for each tuple. Prior to adjudication, annotators achieved an agreement rate of 62% for annotating salient error type, with a Cohen's $\kappa$ of 0.430. Many discrepancies in the first annotation round resulted from confusion between *Malformed Predicate* and *Argument Misattachment* or *Bad Subpredicate*. Often, several error types were present in the incorrect extractions, but deciding which error type was most salient was resolved during adjudication. Examples of each error type and annotation conventions are given in Appendix E.

**OpenIE Results** Results from a typological error analysis of 200 tuples are shown in Table 3. As expected, *Malformed Predicate*s were the predominant source of error for the baseline EWT model, followed by *Missing Core Argument* errors. This agrees with the finding that headline root identification exhibited marked regressions in the baseline.

In our experiments, the domain-specific model was able to drastically reduce errors for both of these error types. We registered a statistically significant improvement (26% absolute) in valid tuple extraction performance when using the output of the model finetuned on EWT+GSC data when compared to the EWT-only baseline. For NYT, the improvement was not statistically significant. We hypothesize that this is due to the fact that the wire exhibits more structural similarities to long-form

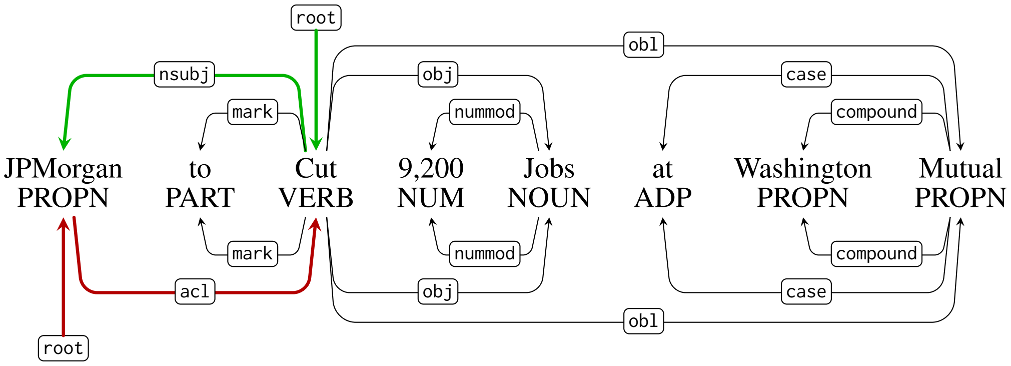


Figure 4: Example parses given by *EWT (baseline)* (bottom) and *Both (finetuned)* (top) on an example headline from the GSC. Differing edges are highlighted in green and red for finetuned and baseline, respectively.

| **Domain** | **Model** | **Malformed Predicate** | **Bad Sub-predicate** | **Missing Core Argument** | **Argument Mis-attachment** | **Incomplete Argument** | **Correct** |
|---|---|---|---|---|---|---|---|
| GSC | EWT | 20 | **4** | 14 | 4 | 2 | 56 |
|  | Finetune | **4**$^{\dagger}$ | 6 | **2**$^{*}$ | 6 | **0** | **82**$^{\dagger}$ |
| NYT | EWT | **10** | 8 | 6 | 12 | **0** | 64 |
|  | Finetune | 12 | **6** | **2** | **4** | 2 | **74** |

Table 3: % error type for OpenIE tuples. Statistically significantly better performance within domain, according to a two population proportion test is indicated by $*$ at the $p = 0.05$ level and † at the $p = 0.01$ level. Sample size of 50 tuples for each (domain, model) pair. Best performing model per (error type, domain) in bold.

text, as evidenced by the frequency of relation types (Figure 2).

## 8 Further Related Work

**Headline Syntactic Processing** Perhaps the two most relevant works are the recently published *POSH* (Benton et al., 2021) and *GoodNewsEveryone* corpora (Oberländer et al., 2020). *POSH* is a dataset of POS-tagged English news headlines, without gold dependency parse annotations. *GoodNewsEveryone*, on the other hand, contains thousands of emotion-bearing headlines labeled for semantic roles (SRL). In *GoodNewsEveryone*, the relationships between identified actor, target, and predicate are solely determined by their roles. Collecting dependency parse annotations is much more involved than either POS tagging or SRL, as dependency parses require identifying deep relationships between individual words that are not solely derived from their types. That said, the release of both of these corpora underscores the importance of headlines as an object of study in NLP, and the desire for richer linguistic annotations.

**Low Resource Syntactic Processing** Low resource syntactic parsing is typically motivated by the need to develop a parser for languages with scant gold supervision (Vania et al., 2019). Agić et al. (2016) employ a similar, yet more involved method of annotation projection to project parser predictions from a high-resource language to a low-resource language. As we are not projecting across languages, and we restrict our parallel text to cases where a headline is a subsequence of the lead sentence, we rely on heuristics to repair the projected dependency parse.

Dependency parsers and treebanks for tweets are similar in spirit to the current work (Owoputi et al., 2013; Kong et al., 2014; Liu et al., 2018). Unlike Tweebank, we chose not to develop our own annotation scheme, but rather annotate under the UD schema. UD is sufficiently expressive for annotating headlines, and allows us to leverage multiple domains for training a parser.

## 9 Conclusion

In this work, we describe the first gold-annotated evaluation set for English headline UD dependency parsing, the EHT. We hope this data will encourage further research in improving dependency parsers for overlooked registers of English. In addition, we hope that the development of accurate headline dependency parsers will result in stronger performance at existing headline understanding and processing tasks, and enable more subtle linguistic analysis, such as identification of “crash blossom” news headlines.

## Limitations

**Variation across news outlets** Figure 3 demonstrates out-of-domain generalization to NYT headlines on several structurally important relations such as `root` and `nsubj:pass` by training on silver projected trees. However, some relations that can be more accurately predicted in GSC headlines do not generalize to NYT. These include adjunct relations such as `nmod`, `nummod`, and `advmod`. Even within a register as niche as English headlinese, there is significant variation in convention between news outlets.

**General news headline distributions** The GSC corpus is originally collected by Filippova and Altun (2013) and contains crawled news headlines and lead sentences from a wide variety of news outlets. The headline-lead-sentence pairs are filtered to include only grammatical and informative headlines (see Section 4 of Filippova and Altun (2013)) and thus the resulting GSC corpus may not be representative of all English news headlines. The NYT corpus contains samples from a single outlet and is also not representative of the general news headline distribution.

**Different news headline categories** Depending on the types of news articles (*e.g.*, front page, editorials, op-eds, *etc.*), their corresponding headlines may exhibit distinctive structural properties. Our work is agnostic to the different categories of news articles and their headlines.

**Multilinguality** In this work, we demonstrate that training on silver parse trees projected onto English news headlines results in more accurate English headline parsers. For other languages, headlines may or may not exhibit significant grammatical differences from English headlines, and when they do, the types of headline constructions are language- and culture-dependent. We expect the benefits of training on projected trees to be mediated by the discrepancy between the “conventional” and headline grammar within a given language. In addition, for languages with richer morphology, morphological analysis may be required to align dependency relation annotations from a body sentence to its headline. As we only consider English headlines in this work, further exploration is required before determining whether the projection algorithm, Algorithm 1, can be adapted to morphologically rich languages.

## Acknowledgements

We thank the anonymous reviewers for their insightful reviews. Tianze Shi acknowledges support from Bloomberg’s Data Science Ph.D. Fellowship.

## References

Isaac Afful. 2014. A diachronic study of the NP structure in Ghanaian newspaper editorials. *Journal of Advances in Linguistics*, 5:555–565.

Željko Agić, Anders Johannsen, Barbara Plank, Héctor Martínez Alonso, Natalie Schluter, and Anders Søgaard. 2016. Multilingual projection for parsing truly low-resource languages. *Transactions of the Association for Computational Linguistics*, 4:301–312.

Xiang Ao, Xiting Wang, Ling Luo, Ying Qiao, Qing He, and Xing Xie. 2021. PENS: A dataset and generic framework for personalized news headline generation. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 82–92, Online. Association for Computational Linguistics.

Joshua Bambrick, Minjie Xu, Andy Almonte, Igor Malioutov, Guim Perarnau, Vittorio Selo, and Iat Chong Chan. 2020. NSTM: Real-time query-driven news overview composition at Bloomberg. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics: System Demonstrations*, pages 350–361, Online. Association for Computational Linguistics.

Adrian Benton, Hanyang Li, and Igor Malioutov. 2021. Cross-register projection for headline part of speech tagging. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 6475–6490, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

Gosse Bouma, Djamé Seddah, and Daniel Zeman. 2020. Overview of the IWPT 2020 shared task on parsing into enhanced Universal Dependencies. In *Proceedings of the 16th International Conference on Parsing Technologies and the IWPT 2020 Shared Task on Parsing into Enhanced Universal Dependencies*, pages 151–161, Online. Association for Computational Linguistics.

Gosse Bouma, Djamé Seddah, and Daniel Zeman. 2021. From raw text to enhanced Universal Dependencies: The parsing shared task at IWPT 2021. In *Proceedings of the 17th International Conference on Parsing Technologies and the IWPT 2021 Shared*

*Task on Parsing into Enhanced Universal Dependencies (IWPT 2021)*, pages 146–157, Online. Association for Computational Linguistics.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4171–4186, Minneapolis, Minnesota. Association for Computational Linguistics.

Timothy Dozat and Christopher D. Manning. 2017. Deep biaffine attention for neural dependency parsing. In *Proceedings of the 5th International Conference on Learning Representations*, pages 1–8, Toulon, France. OpenReview.net.

Taiwo Oluwaseun Ehineni. 2014. A syntactic analysis of lexical and functional heads in nigerian english newspaper headlines. *International Journal of Linguistics*, 6(5):9.

William Ferreira and Andreas Vlachos. 2016. Emergent: a novel data-set for stance classification. In *Proceedings of the 2016 conference of the North American chapter of the association for computational linguistics: Human language technologies*, pages 1163–1168.

Katja Filippova and Yasemin Altun. 2013. Overcoming the lack of parallel data in sentence compression. In *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing*, pages 1481–1491.

Rama Rohit Reddy Gangula, Suma Reddy Duggenpudi, and Radhika Mamidi. 2019. Detecting political bias in news articles using headline attention. In *Proceedings of the 2019 ACL Workshop BlackboxNLP: Analyzing and Interpreting Neural Networks for NLP*, pages 77–84.

Robert E Garst and Theodore M Bernstein. 1933. *Headlines and deadlines.* Columbia University Press.

Diederik P. Kingma and Jimmy Ba. 2015. Adam: A method for stochastic optimization. In *Proceedings of the 3rd International Conference on Learning Representations*, San Diego, California, USA.

Dan Kondratyuk and Milan Straka. 2019. 75 languages, 1 model: Parsing Universal Dependencies universally. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 2779–2795, Hong Kong, China. Association for Computational Linguistics.

Lingpeng Kong, Nathan Schneider, Swabha Swayamdipta, Archna Bhatia, Chris Dyer, and Noah A Smith. 2014. A dependency parser for tweets. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1001–1012.

Zornitsa Kozareva, Borja Navarro, Sonia Vázquez, and Andrés Montoyo. 2007. Ua-zbsa: a headline emotion classification through web information. In *Proceedings of the Fourth International Workshop on Semantic Evaluations (SemEval-2007)*, pages 334–337.

Philippe Laban, Lucas Bandarkar, and Marti A Hearst. 2021. News headline grouping as a challenging nlu task. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 3186–3198.

Siyi Liu, Lei Guo, Kate Mays, Margrit Betke, and Derry Tanti Wijaya. 2019. Detecting frames in news headlines and its application to analyzing news framing trends surrounding us gun violence. In *Proceedings of the 23rd Conference on Computational Natural Language Learning (CoNLL)*, pages 504–514.

Yijia Liu, Yi Zhu, Wanxiang Che, Bing Qin, Nathan Schneider, and Noah A Smith. 2018. Parsing tweets into universal dependencies. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pages 965–975.

Ingrid Mårdh. 1980. *Headlinese: On the grammar of English front page headlines*, volume 58. Liberläromedel/Gleerup.

Laura Ana Maria Oberländer, Evgeny Kim, and Roman Klinger. 2020. Goodnewseveryone: A corpus of news headlines annotated with emotions, semantic roles, and reader perception. In *Proceedings of the 12th Language Resources and Evaluation Conference*, pages 1554–1566.

Olutobi Owoputi, Brendan O'Connor, Chris Dyer, Kevin Gimpel, Nathan Schneider, and Noah A Smith. 2013. Improved part-of-speech tagging for online conversational text with word clusters. In *Proceedings of the 2013 conference of the North American chapter of the association for computational linguistics: human language technologies*, pages 380–390.

Peng Qi, Yuhao Zhang, Yuhui Zhang, Jason Bolton, and Christopher D. Manning. 2020. Stanza: A Python natural language processing toolkit for many human languages. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics: System Demonstrations*, pages 101–108, Online. Association for Computational Linguistics.

Alexander M Rush, Sumit Chopra, and Jason Weston. 2015. A neural attention model for abstractive sentence summarization. In *Proceedings of the 2015*

*Conference on Empirical Methods in Natural Language Processing*, pages 379–389.

Kenji Sagae and Alon Lavie. 2006. Parser combination by reparsing. In *Proceedings of the Human Language Technology Conference of the NAACL, Companion Volume: Short Papers*, pages 129–132, New York City, USA. Association for Computational Linguistics.

Kristina Schneider. 2000. The emergence and development of headlines in british newspapers. *English Media Texts, Past and Present: Language and Textual Structure*, 80:45.

Allan M Siegal and William G Connolly. 1999. *The New York Times manual of style and usage*. Three Rivers Press (CA).

Natalia Silveira, Timothy Dozat, Marie-Catherine de Marneffe, Samuel Bowman, Miriam Connor, John Bauer, and Christopher D. Manning. 2014. A gold standard dependency corpus for English. In *Proceedings of the Ninth International Conference on Language Resources and Evaluation (LREC-2014)*.

Carlo Strapparava and Rada Mihalcea. 2007. Semeval-2007 task 14: Affective text. In *Proceedings of the Fourth International Workshop on Semantic Evaluations (SemEval-2007)*, pages 70–74.

Heinrich Straumann. 1935. *Newspaper headlines: A study in linguistic method*. London, Allen.

Sara Stymne, Miryam de Lhoneux, Aaron Smith, and Joakim Nivre. 2018. Parser training with heterogeneous treebanks. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 619–625, Melbourne, Australia. Association for Computational Linguistics.

Sho Takase and Naoaki Okazaki. 2019. Positional encoding to control output sequence length. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 3999–4004.

Sho Takase, Jun Suzuki, Naoaki Okazaki, Tsutomu Hirao, and Masaaki Nagata. 2016. Neural headline generation on abstract meaning representation. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 1054–1059.

Jiwei Tan, Xiaojun Wan, and Jianguo Xiao. 2017. From neural sentence summarization to headline generation: a coarse-to-fine approach. In *Proceedings of the 26th International Joint Conference on Artificial Intelligence*, pages 4109–4115.

Francis M. Tyers, Mariya Sheyanova, and Jonathan North Washington. 2017. UD Annotatrix: An annotation tool for Universal Dependencies. In *Proceedings of the 16th International Workshop on Treebanks and Linguistic Theories*, pages 10–17, Prague, Czech Republic.

AM Simon Vanderbergen. 1981. *The Grammar of Headlines in the Times: 1870-1970*, volume 95. AWLSK.

Clara Vania, Yova Kementchedjhieva, Anders Søgaard, and Adam Lopez. 2019. A systematic comparison of methods for low-resource dependency parsing on genuinely low-resource languages. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 1105–1116.

Aaron Steven White, Drew Reisinger, Keisuke Sakaguchi, Tim Vieira, Sheng Zhang, Rachel Rudinger, Kyle Rawlins, and Benjamin Van Durme. 2016. Universal Decompositional Semantics on Universal Dependencies. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 1713–1723, Austin, Texas. Association for Computational Linguistics.

Rachel Wities, Vered Shwartz, Gabriel Stanovsky, Meni Adler, Ori Shapira, Shyam Upadhyay, Dan Roth, Eugenio Martínez-Cámara, Iryna Gurevych, and Ido Dagan. 2017. A consolidated open knowledge representation for multiple texts. In *Proceedings of the 2nd workshop on linking models of lexical, sentential and discourse-level semantics*, pages 12–24.

David Yarowsky, Grace Ngai, and Richard Wicentowski. 2001. Inducing multilingual text analysis tools via robust projection across aligned corpora. In *Proceedings of the First International Conference on Human Language Technology Research*.

Amir Zeldes. 2017. The GUM corpus: Creating multilayer resources in the classroom. *Language Resources and Evaluation*, 51(3):581–612.

# A Syntactic Annotation Guidelines

## General principles

Please refer to the UD annotation guidelines (https://universaldependencies.org/guidelines.html) for general rules of syntactic dependency annotation. This document serves as an addendum to the UD guidelines, in order to detail how to annotate certain frequently occurring and/or headline-specific constructions.

## Frequent headline constructions

**Headlines with multiple decks/components** Use parataxis to connect multiple components. For example:

(1) Paid Notice: Deaths BROOKS, JOHN N.

includes three independent components: ”Paid Notice”, ”Deaths”, and ”BROOKS, JOHN N.”, with the latter two attached to the first through parataxis. There can be nested parataxis if necessary to reflect hierarchical structures within the headline components.

**Headlines with omitted auxiliaries** For frequent constructions including ”NP VP$_{ed}$”, ”NP VP$_{ing}$”, and ”NP VP$_{to}$”, where the finite auxiliary ”be” verbs are omitted, we still treat the headlines as verbal headlines and mark the main verbs as the root/head of the headlines.

**Reported speech** Refer to the UD guidelines. Typically, a `ccomp` or `parataxis` relation is used.

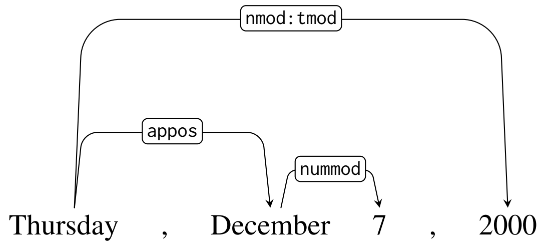
**`flat` and `compound`** First, refer to UD guidelines on `flat` and `compound`. These are typically annotated as flat:

- (Person) Names
- Company/team/organization/... names without internal (compositional) structures. (e.g., ”Rolling Stones” is not compositional and should be analyzed as flat.)
- Foreign phrases
- Dates without explicit internal structures (excluding ”the 1st of May”)
- Titles/honorifics

## Dates

(2) Thursday, December 7, 2000

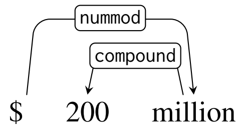
Refer to English web treebank example `email-enronsent06_01-0005`



## Currency

(3) $ 200 million

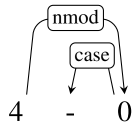
Refer to English web treebank example: `newsgroup-groups.google.com_FOOLED_1bf9cdc5a4c2ac48_ENG_20050904_130400-0022`



## Game Scores

(4) 4 - 0

Refer to English web treebank example: `newsgroup-groups.google.com_hiddennook_1fd8f731ae7ffaa0_ENG_20050214_192900-0006`



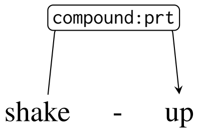
## Special cases

**Named Entities** Locations should be annotated similarly to people names, with `flat`. Therefore ”Lake Erie” should be parsed using `flat` rather than `compound`. Although this conflicts with how locations are annotated in EWT, the judgments in EWT are occasionally inconsistent or conflict with the UD annotation guidelines, as evidenced by UD issue 777 – https://github.com/UniversalDependencies/docs/issues/777. `flat` should also be used for names of racing horses, where although there is often compositional structure in these names, they are treated as a single unit as there are loose syntactic constraints on what constitutes a valid race horse name.

Company names with typical suffixes like ”Inc.” or ”Co.” should be analyzed with that word as the head, with a compound relation to the idiosyncratic part of the company name. Arbitrary names of companies should be analyzed with flat. Names of creative works: visual art, books, movies, video games should be annotated such that internal structure is preserved, e.g., ”Lord of the Rings” is not parsed with flat.

**Legitimate PP Attachment Ambiguity** In certain cases there may be inherent ambiguity in where a prepositional phrase should attach, but the syntactic ambiguity has little effect on the meaning of the headline (nmod on an oblique/object argument vs. obl attaching to the matrix verb). In these cases, we chose to attach the PP as an nmod to the argument, out of convention.

**Hyphenated Words** In the case of hyphenated words, we analyze the internal structure of the hyphenated words and attach as the entire hyphenated word functions in the headline. For example, ”shake-up” is parsed as:



even if the entire word functions as a noun.

**Typos** In the case of typographical errors or issues with data processing, we assume the intended word during annotation. So, for ”Baby game changer for to”, ”to” is labeled as ”NUM”, assuming ”two” was the intended word. These typographical errors will be remedied by their corrected lemma in the future.

## B Implementation of the Projection Algorithm

Figure 5 provides a detailed python implementation of Algorithm 1 for reproducibility.

## C Parser and Implementation Details

Our parser architecture combines the deep biaffine parser (Dozat and Manning, 2017) with the pre-trained contextual BERT feature extractor (Devlin et al., 2019). For words with multiple subword tokens, we adopt BERT representations on the final subword tokens. For the deep biaffine parser, the attachment and labeling probabilities are determined by biaffine attention scores between pairs of head-dependent words, which in turn are linearly projected from BERT embeddings and then followed by a non-linear leaky ReLU activation function. We used a dimension of 400 for the attachment biaffine scorer, and 100 for the label scorer. For the BERT feature extractor, the weights are initialized from the public bert-base-uncased model,[11] consisting of roughly 110 million parameters, and fine-tuned during training. Each model was trained on a single Nvidia GTX 2080 Ti GPU, and took up to two hours to train depending on when training was halted.

We selected learning rate on the baseline EWT model,[12] and used the same hyperparameter settings when training all other parsers. We used a maximum learning rate of $10^{-5}$, a batch size of 8, and a learning rate schedule that tenths the base learning rate every 5 iterations without increase in validation accuracy, up to two times maximum. Learning rate was warmed up according to a linear schedule during the first 320 iterations. Gradients are clipped to a maximum norm of 5.0. We used Adam optimizer (Kingma and Ba, 2015) with $\beta_1 = 0.9$, $\beta_2 = 0.999$ for all training runs. For fine-tuning, we used a maximum learning rate of $10^{-6}$, with an identical learning rate schedule. Dropout rates of 0.3 are applied to all non-linear activations in the parsing modules.

## D Intrinsic Performance by Relation

Per-relation absolute F1 is displayed Figure 6 for the ensembled baseline EWT-trained parser vs. additionally finetuning on projected GSC trees.

## E OpenIE Annotation Details

Table 4 contains a handful of examples for each of the salient OpenIE error types annotated in Section 7. Please refer to Benton et al. (2021) for descriptions of each of these error types. In addition to that protocol, we adopted the following annotation conventions in order to consistently annotate corner cases. In general, tuples were labeled as incorrect *only* if there were clear mistakes in the definition of arguments or predicate:

- Tuples where the substructure of the reported

[11] https://huggingface.co/bert-base-uncased

[12] From the set of $\{5 \times 10^{-6}, 10^{-5}, 2 \times 10^{-5}, 5 \times 10^{-5}\}$.

```python
def project(heads: List[int], rels: List[str], subset: List[int]):
  """Finds the subtree spanned by the nodes in `subset`.

  Arguments:
    heads: Each element `heads[i]` corresponds to the head of node `i`;
      node 0 is root, and `heads[0] = -1`.
    rels: Each element `rels[i]` is the dependency relation label between
      `heads[i]` and node `i`.
    subset: A list of nodes that are found in the subtree.

  Returns:
    A tuple of heads and relations in the same format as `heads` and `rels`,
    the length of each is equal to `len(subset)`.
  """
  heads = deepcopy(heads)
  rels = deepcopy(rels)

  # Collect all involved nodes (ExtractSubtree).
  included = set()
  for i in subset:
    cur = i
    while cur != -1:
      included.add(cur)
      cur = heads[cur]

  # Cache the children of each node.
  children = [[] for x in heads]
  for i in sorted(included):
    if heads[i] != -1:
      children[heads[i]].append(i)

  while len(included) != len(subset):
    # Find the top-most node that is not currently in the subset.
    queue = [-1]
    while len(queue):
      cur = queue.pop()
      if cur not in subset and cur != -1:
        node_to_collapse = cur
        break
      queue.extend(children[cur])

    # Find the local structure and collapse.
    children_nodes = children[node_to_collapse]
    leftmost = children_nodes[0]
    for c in children_nodes:
      heads[c] = leftmost
    heads[leftmost] = heads[node_to_collapse]
    rels[leftmost] = rels[node_to_collapse]
    included.discard(node_to_collapse)

    # Update cache.
    children = [[] for x in heads]
    for i in sorted(included):
      if heads[i] != -1:
        children[heads[i]].append(i)

  # Extract the subgraph.
  mapping = {n: i for i, n in enumerate(subset)}
  subset_heads = [mapping.get(heads[x] , -1) for x in subset]
  subset_rels = [rels[x] for x in subset]

  return subset_heads, subset_rels
```

Figure 5: The python implementation of Algorithm 1.

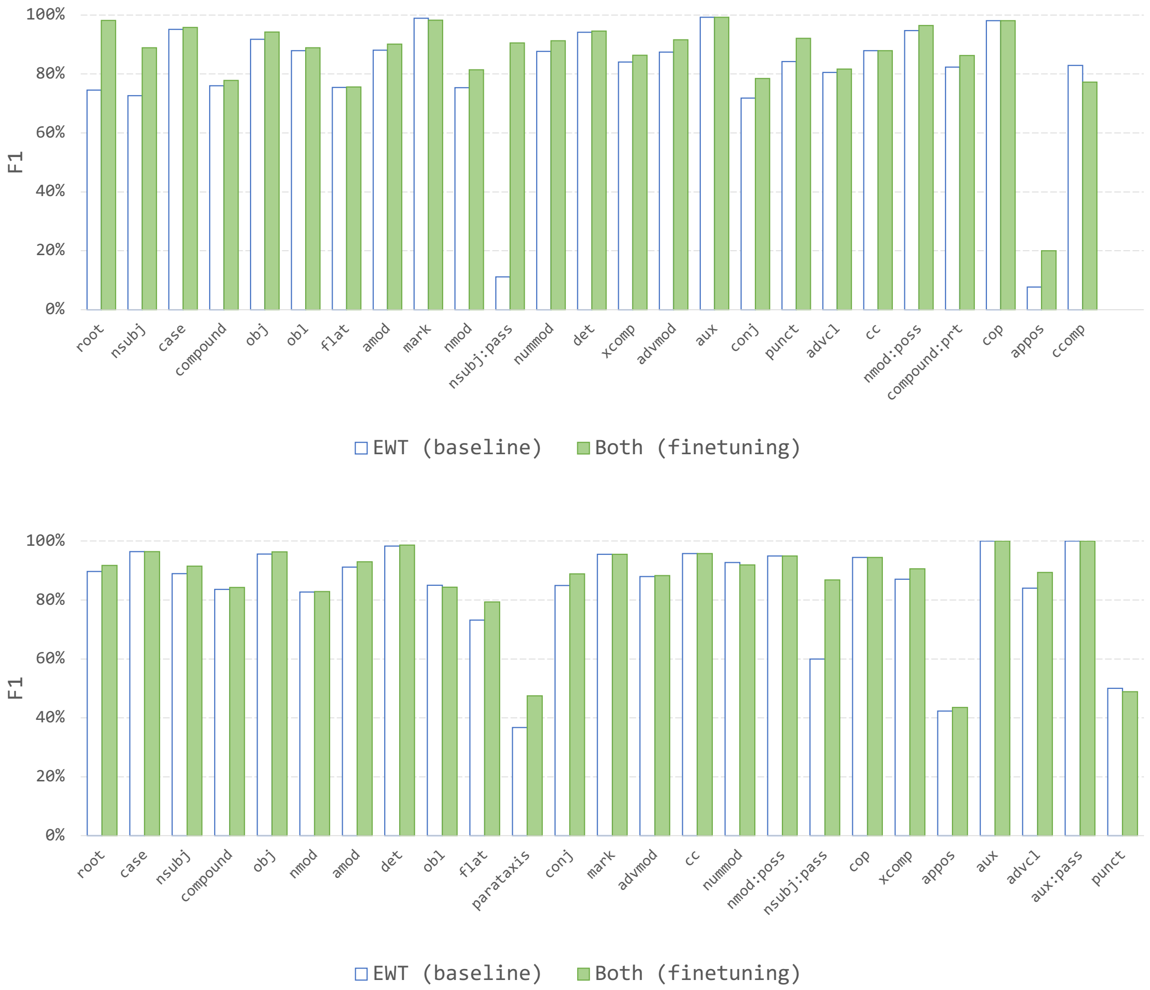


Figure 6: % F1 score across dependency relations for both the GSC (top) and NYT (bottom) evaluation sets for the ensembled EWT-only model against the finetuned EWT+projected GSC predictions. Relations are sorted by descending frequency and only relations that occurred at least 20 times in the evaluation set are shown.

phrase is decomposed as additional arguments in reporting structures ("Prime Minister says...") are judged "Correct".

- A complicated predicate was labeled as valid, even when an object could have been treated as a separate argument.
- Tuples of independent decks, related by parataxis, or appositives are judged "Correct".
- Sub-predicates that are entailed by the headline are judge "Correct". For example, "X engaged to wed" → (wed, X) ; (engaged, X) ; (engaged to wed, X) are all valid.
- Relative pronouns should not be included as separate arguments in the relative clause, as they are redundant with the nominal head.

| **Malformed Predicate** |
|---|
| [$_{A1}$ *Torrid heatwave sweeps*] [$_{P}$ **Punjab**]<br>[$_{P}$ **Kenya acrobat falls during**] [$_{A1}$ *circus show in Moscow*]<br>[$_{P}$ **Will Wright to leave**] [$_{A1}$ *Electronic Arts*] |
| **Missing Core Argument** |
| Toyota to [$_{P}$ **revise**] [$_{A1}$ *dollar forecast to 80 yen*]<br>Several paths available to [$_{P}$ **extend**] [$_{A1}$ *the litigation*] |
| **Bad Sub-Predicate** |
| [$_{A1}$ *Sanofi*] to[$_{P}$ **take**] [$_{A2}$ *control of Shantha Biotechnics*] |
| **Argument Misattachment** |
| [$_{A1}$ *J.*] [$_{A2}$ *W. Kirby*] [$_{P}$ **wed to**] [$_{A2}$ *miss McCabe*]<br>[$_{A1}$ *Bishop*] [$_{A2}$ *who*] [$_{P}$ **had denied**] [$_{A2}$ *Holocaust*] apologizes |
| **Incomplete Argument** |
| A raft of [$_{A1}$ *plans*] [$_{A2}$ *that*] [$_{P}$ **try to dispel**] [$_{A2}$ *math anxieties*] |

Table 4: Example salient error types for OpenIE tuples extracted from the baseline EWT-only ensemble parse. Extracted tuples encoded as $\langle$[$_{P}$ **Predicate**], [$_{A1}$ *Argument 1*], [$_{A2}$ *Argument 2*]$\rangle$.