A Neural Morphological Analyzer for Arapaho Verbs Learned from a Finite State Transducer

Sarah Moeller and **Ghazaleh Kazeminejad** and **Andrew Cowell** and **Mans Hulden**
Department of Linguistics
University of Colorado
first.last@colorado.edu

Abstract

We experiment with training an encoder-decoder neural model for mimicking the behavior of an existing hand-written finite-state morphological grammar for Arapaho verbs, a polysynthetic language with a highly complex verbal inflection system. After adjusting for ambiguous parses, we find that the system is able to generalize to unseen forms with accuracies of 98.68% (unambiguous verbs) and 92.90% (all verbs).

1 Introduction

One of the clear successes in computational modeling of linguistic patterns has been that of finite state transducer (FST) models for morphological analysis and generation (Koskenniemi, 1983; Beesley and Karttunen, 2003; Hulden, 2009; Lindén et al., 2009). Given enough linguistic expertise and investment in developing such models, FSTs provide the capability to analyze any well-formed word in a language. Although FST models generally rely on lexicons, they can also be extended to handle complex inflected word forms outside the lexicon as long as morphophonological regularities are obeyed. Even ill-formed words can be mapped to a "closest plausible reading" through FST engineering (Beesley and Karttunen, 2003). On the downside, developing a robust FST model for a given language is very time-consuming and requires knowledge of both the language and finite-state modeling tools (Maxwell, 2015). Development of a finite-state grammar tends to follow a Pareto-style tradeoff where the bulk of the grammar can be developed very quickly, and the long tail of remaining effort tends to focus on lexicon expansion and difficult corner cases.

In this paper we describe an experiment in training a neural encoder-decoder model to replicate the behavior of an existing morphological analyzer for the Arapaho language (Kazeminejad et al., 2017). Our purpose is to evaluate the feasibility of bootstrapping a neural analyzer with a hand-developed FST grammar, particularly if we train from an incomplete selection of word forms in the hand-developed grammar. A successful morphological analyzer is essential for downstream applications, such as speech recognition and machine translation, that could provide Arapaho speakers access to common tools similar to Siri or Google Translate that might support and accelerate language revitalization efforts.

2 Background & Related Work

Neural network models for word inflection have increased in popularity, particularly following the two SIGMORPHON and CoNLL-SIGMORPHON shared tasks (Cotterell et al., 2016; Cotterell et al., 2017). Most of the work in this domain relies on training encoder-decoder models used in machine translation to perform 'translations' of base forms and grammatical specifications into output forms, such as `fly +V +3PPres` ↦ `flies`, or vice versa. While such models can produce very reliable systems with a few thousand examples, the small available sample of polysynthetic languages indicate they are slightly more difficult to learn. Compare, for example, the accuracies of the best teams at CoNLL-SIGMORPHON 2017 between Navajo (92.30%) and Quechua[1] (99.90%). A remarkable detail about the neural inflection

[1]An agglutinating language with complex morphology, though not considered polysynthetic.

Proceedings of Workshop on Polysynthetic Languages, pages 12–20
Santa Fe, New Mexico, USA, August 20-26, 2018.

models is that in the 2017 shared task they were found to generalize beyond feature combinations that they had witnessed. Thus, for example, if a system had seen future tense forms and plurals separately, but never seen the combination of the two, they could produce the combination quite reliably (Cotterell et al., 2017). This effect was most striking for Basque, which has a highly complex, albeit very regular, verb system. One of the main purposes of the experiments described in this paper is to capitalize on this capacity to automatically generalize beyond what has been explicitly encoded in an FST grammar.

Standard morphological analyzers tend to be designed to return all 'plausible' parses of a word. In English, for example, this means that in practice any verb (e.g. **sell**) would always be alternatively parsed as a noun reading as well; likewise for the third person present form, **sells**, which could be parsed as a plural noun. This adds complications to the design of a neural model intended to mimic the behavior of a classical morphological analyzer, since it needs to return multiple options, and a neural encoder-decoder really encapsulates a distribution over all possible output strings Σ^* for any input string read by the encoder. An unexpected "advantage" of applying this to polysynthetic languages is that, while the verb complex in polysynthetic languages tends to be very intricate and is time-consuming to model, it proffers typically less ambiguity of a parse (as will be discussed in Section 6). Even when ambiguous readings are possible, they tend to be highly systematic. Silfverberg and Hulden (2018) documents a neural model from an FST model for Finnish (an agglutinative language) to retrieve all plausible parses of a word form, reporting an F_1-score of 0.92. The authors report that the recall was far lower than the precision, indicating difficulty in learning to return all the valid parses. The problem of unsystematic ambiguity, however, can often be avoided in the parsing of verbs in polysynthetic languages with mostly systematic ambiguity. Navajo, for example, collapses singulars and duoplurals in the 3rd and 4th person, and so the ambiguity between the two could be encoded by introducing an additional super-tag representing both options at once.[2] In other words, systematic multiple readings can be circumvented in systems designed to give a single parse by simply altering the tagset for relevant cases, such that a parse with the tag [`SG/DPL`] could be interpreted as a two-way ambiguity. Another example is seen in Algonquian languages, which often have homophonous participle forms of verbs—affixes expressing features of the possessor are often homophonous with affixes expressing features of the subject or object.[3]

3 The Arapaho Verb

Arapaho is a member of the Algonquian (and larger Algic) language family; it is an agglutinating, polysynthetic language, with free word order (Cowell and Moss Sr, 2008). The language has a very complex verbal inflection system, with a number of typologically uncommon elements. Verb stems have unique stem-final elements which specify for valency and animacy: a given stem is used either with animate or inanimate subjects for intransitive verbs (**tei'<u>eihi</u>-** 'be strong.animate' vs. **tei'<u>oo</u>-** 'be strong.inanimate'), and with animate or inanimate objects for transitive verbs (**nooh<u>ow</u>-** 'see s.o.' vs. **nooh<u>oot</u>-** 'see s.t.'). For each of these categories of stems, the pronominal affixes/inflections that attach to the verb stem vary in form, for example, 2SG with intransitive, animate subject verbs is **/-n/**, while for transitive, inanimate object verbs it is **/-ow/** (**nih-tei'eihi-n** 'you were strong' vs. **nih-noohoot-ow** 'you saw it').

All of these stem types can occur in four different verbal orders, whose function is primarily modal—affirmative, conjunct/subordinate, imperative, and non-affirmative. These verbal orders each use different pronominal affixes/inflections as well. For example, when a verbal root such as **/nooh-/** 'see' is transitive with an animate object, 2SG acting on 3SG is **/-in/** or **/-un/** for the imperative order (**noohow-un** '(you) see him!'), but **/-ot/** for the affirmative order (**nih-noohow-ot** 'you saw him'), and with the non-affirmative order the 2SG marker is a prefix, **/he-/**, not a suffix. Thus, with four different verb stem types and four different verbal orders, there are a total of 16 different potential inflectional paradigms for any verbal root, though there is some overlap in the paradigms, and not all stem forms are possible for all roots.

[2] The fourth person in Navajo is the form for an obligatorily human "impersonal" third person participant (Akmajian and Anderson, 1970; Young and Morgan, 1987).

[3] Thank you to an anonymous reviewer for this example.

Two final complications are vowel harmony with related consonant mutation, and a proximate/obviative system. Arapaho has both progressive and regressive vowel harmony, operating on **/i/** and **/e/** respectively. This results in alternations in both the inflections themselves, and the final elements of stems, such as **noohow-un** 'see him' vs. **niiteheib-in** 'help him', or **nih-ni'eeneb-e3en** 'I liked you' vs. **nih-ni'eenow-oot** 'he liked her'. Arapaho also has a proximate/obviative system, which does not overlap with either subject/object or agent/patient categories, but instead designates pragmatically more- and less-prominent participants. There are "direction-of-action" markers (elsewhere, for simplicity, we use "subject" and "object") included in inflections, which do not correspond to true pronominal affixes. Thus **nih-noohow-oot** 'more important 3SG saw less important 3S/PL' vs. **nih-noohob-eit** 'less important 3SG/PL saw more important 3S', and **nih-noohob-einoo** 'less important 3S saw more important 1S'. The elements **-oo-** and **-ei-** specify direction of action, not specific persons or numbers of participants. some of these suffixes produce systematic ambiguity, as shown in Table

Some "direction-of-action" markers generate ambiguity in person and number of the verb's arguments. Thus, for example, in **nih-noohob-eit** 'less important 3SG/PL saw more important 3SG' the **/-eit/** suffix is systematically ambiguous as to the number of the less important/obviative 3rd-person participant.

4 Finite-State Model

A morphological analyzer is a prerequisite for many NLP tasks. It is even more crucial to have such a parser for morphologically complex languages such as Arapaho. A finite state transducer (FST) is the standard technology for creating morphological analyzers. The FST is bidirectional and able to simultaneously parse given inflected word forms and generate all possible word forms for a given stem (Beesley and Karttunen, 2003).

The Arapaho FST model used in this paper was constructed with the *foma* finite-state toolkit (Hulden, 2009). The FST is constructed in two parts, the first being a specification of the lexicon and morphotactics using the finite-state lexicon compiler (lexc). This is a high-level declarative language for effective lexicon creation, where concatenative morphological rules and morphological irregularities are addressed (Karttunen, 1993).

The second part implements the morphophonological rules of the language using "rewrite rules" that apply the appropriate changes in specified contexts. This way, the generated inflected word form is not merely a bundle of morphemes, but the completely correct form in accord with the morphophonological rules of the language. So, by applying, in a particular order (specified in the grammar of the language), the rewrite rules to the parsed forms generated in the lexc file, the result is a single FST able to both generate and parse. Figure 1 shows how the FST is designed to generate and parse an example.

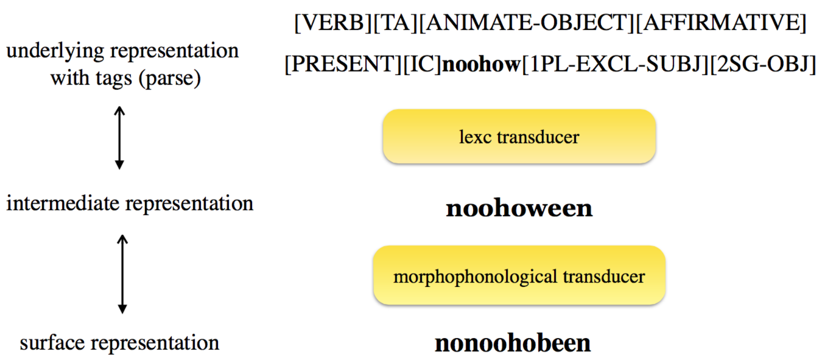

Figure 1: Composition in an FST illustrating the underlying (input) parsed forms and the resulting surface (output) inflected forms after mapping morpheme tags to concrete morphemes and subsequently undergoing morphophonological alternations.

The generalized application of rewrite application such that a FST or a neural model based on an

FST which is described below may seem like a "manufacturing" of the language, applying grammatical rules to verbal stems in order to create artificial forms. However, a morphological analyzer works with inflections and various kinds of prefixes; it does not build new verb stems. For the most part, it is the stems themselves that encode culture-sensitive information and perspectives.

5 Training a recurrent neural network from an FST

Tag	Description	Tag	Description
1PL-EXCL	First Person Plural Exclusive	1PL-INCL	First Person Plural Inclusive
3	Third Person Proximate	4	Third Person Obviative
II	Inanimate Subject Intransitive Verb	AI	Animate Subject Intransitive Verb
TI	Inanimate Object Transitive Verb	TA	Animate Object Transitive Verb
IC	Initial Change		

Table 1: Description of non-self-explanatory tags used in the parser

For training data, we extract inflected word forms and their corresponding parsed forms generated by the FST model for Arapaho, i.e. pairs such as the one seen at the top of Figure 1. These pairs serve as supervised examples to train a recurrent neural network (RNN) encoder-decoder. The set of FST-extracted input-output pairs contains 584,574 examples; however, a few forms had been incorrectly stored in the database and had to be identified and filtered before training. We trained on 60% of the filtered pairs. Another 20% was withheld for validation and an additional 20% as the test set. We evaluate the ability of the RNN to provide correct parses to unseen inflected word forms.

Since the currently strongest performing models for the related task of morphological inflection (Cotterell et al., 2017; Kann et al., 2017; Makarov et al., 2017) use an LSTM-based sequence-to-sequence (seq2seq) model (Sutskever et al., 2014), we follow this design in our work. Following Kann et al. (2017) who found that adding an attention mechanism (Bahdanau et al., 2015) improved performance, we always include attention as well. We treat parsing as a translation task of input character sequences from the fully-inflected surface forms to an output sequence of morphosyntactic tags plus the character sequences of the verbal root, i.e. we treat the root as the citation form to be retrieved. We implement the bidirectional LSTM-based sequence to sequence model with OpenNMT (Klein et al., 2017), using the default parameters that employ 2 layers for both the encoder and decoder, a hidden size of 500 for the recurrent unit, and a maximum batch size of 64. We train the model until the perplexity converges (at 1.02 or 1.01 for ambiguous and combined data, and 1.00 for unambiguous data), which usually occurs within 5 epochs and generally does not improve significantly with additional epochs. We experimented adding additional layers but without noticeable difference in the results.

As previous authors (Sutskever et al., 2014) have documented a sensitivity to element ordering, we experimented with training the model using various relative orders of morphosyntactic tags and the root morpheme: `Tags+Root`, `Root+Tags`, `Tags+Root+Tags`. These orders are shown in Table 2. (Table 1 provides the description of tags used in the parser that may not be self-explanatory).

Only the `Tags+Root` order was able to produce a model that parses any single inflected form completely correct. Examining the results of the `Tags+Root` predictions revealed that a majority of the mistakes involve the final letter of the root. The model often incorrectly predicts the last letter of the root morpheme, leaves it out completely, or adds an additional letter. Using an end-of-sequence marker does not affect this tendency, which we did not investigate further as we were able to avoid its effect by simply altering the order of the `Tags+Root` elements and the evaluation process. First, we trained the model with a `Tags+Root+Tags`[4] order, duplicating the morphosyntactic tags on both sides or the root, in the order that they were generated by the FST. After training, we removed the set of tags following the root and evaluated the neural encoder-decoder's predictions only against `Tags+Root` ordering of the test set.

[4]Repeating the Tags

Order	Example
Tags+Root	[VERB][TA][ANIMATE-OBJECT][AFFIRMATIVE][PRESENT][IC][1PL-EXCL-SUBJ][2SG-OBJ]**noohow**
Root+Tags	**noohow**[VERB][TA][ANIMATE-OBJECT][AFFIRMATIVE][PRESENT][IC][1PL-EXCL-SUBJ][2SG-OBJ]
Tags+Root+Tags	[VERB][TA][ANIMATE-OBJECT][AFFIRMATIVE][PRESENT][IC][1PL-EXCL-SUBJ][2SG-OBJ]**noohow**[VERB][TA][ANIMATE-OBJECT][AFFIRMATIVE][PRESENT][IC][1PL-EXCL-SUBJ][2SG-OBJ]

Table 2: Examples of orders of morphosyntactic tags and roots used for training the neural model. For encoder-decoder training, spaces were placed between square brackets and individual letters of the root. Thus, tags and letters were treated as single units for "translation".

Once high accuracy was reached on inflected word forms with only one possible parse, the ambiguous wordforms were added to the data. With no adjustments made to the output of the FST, the model parsed 46% of the test data completely correct. Removing all ambiguous surface forms which have than one possible parse increased the accuracy to 60%. With this setup, the accuracy for parsing full words did not exceed 60% without adjustments made for ambiguous words, the overall F_1-scores on individual tags and characters averaged over 0.9, indicating that, although 40% of the predicted parses contained at least one mistake, very few mistakes were made per wordform. Ambiguous forms were "disambiguated" for parsing by altering the tagset. Multiple morphosyntactic tags that are generated by one morpheme became a single tag containing generic information. For example, the word **nih-noohob-e̲it** 'less important 3SG/PL saw more important 3SG' has two possible parses. Its **/-eit/** suffix is systematically ambiguous as to the number of the less important/obviative 3rd-person participant. So the tagset substituted the two alternative parses—[3SG-SUBJ][3SG-OBJ] or [3PL-SUBJ][3SG-OBJ]—with a single new tag [3-SUBJ.3SG-OBJ]. Altering the tagset like this makes the predicted parsed forms less informative, since morphosyntactic information is lost for the sake of generalization. However, the predicted parses are no less ambiguous than are the corresponding fully-inflected Arapaho words when removed from context.

6 Results

Ambiguous and unambiguous word forms combined produce a training data size of about 245,600 supervised pairs. A little over half of those have unambiguous parses, but the actual percentage of unambiguous forms proffered by Arapaho's polysynthetic verbal inflection is probably closer to 75% because repeated ambiguous forms were not eliminated from the data. The RNN model was trained to produce root morphemes and morphosyntactic tags from fully-inflected word forms. The most accurate results came from training the model with morphosyntactic tags repeated before and after the root morpheme and removing the final set of tags before evaluating the model's prediction on the test set (Tags+Root+Tags ⇒ Tags+Root). Training only on unambiguous wordforms resulted in a final accuracy of 98.68%. After ambiguous forms were added to the data and the tagset was altered to "disambiguate" systematic alternative parses, the model's accuracy dropped from to 92.90%. This is better than the model's predictions of the ambiguous pairs on their own: 88.06%. The results of the model's prediction on individual tags and letters are broken down in the Appendix.

The nearly 93% accuracy is obtained by minimal disambiguation of ambiguous word forms. We removed specification of person and number from some arguments to account for the ambiguity of "direction-of-action" morphemes. The relatively low scores on certain tags, as shown in the Appendix, indicate that this accounts for only part of Arapaho's verbal morphological ambiguity. Other morphosyntactic information is ambiguous or, at least, more difficult to identify. For example, the difference between some transitive and intransitive verbs. Also, even some of the altered "direction-of-action" tags could be altered to become even less generic. Pre-processing should identify these morphemes and replace the alternative parses with as accurate a super-tag as the language's ambiguity allows. Such further disambiguation is a longer tail for future work, undoubtedly complicated by morphophonemic changes.

7 Discussion

Since even an endangered language expands and changes, a morphological analyzer that generalizes to unseen inflected forms is more useful than one that does not. Handwritten rules cannot reach into the long tail of lexicon expansion and difficult corner cases. The neural encoder-decoder model described in this paper overcomes the limitations of FST and handwritten rules. One advantage of an FST is the large number of surface and parsed pairs it generates for supervised training of our neural model. We paid attention to the best ordering of the morphosyntactic tags and verbal roots in the training data and found the best combination was training on Tags+Root+Tags and evaluating on Tags+Root. Our neural model can generalize with nearly 93% accuracy beyond what is explicitly encoded. This result comes in part from the lack of systematic ambiguity in a polysynthetic language such as Arapaho, but future work should increase the usefulness of the parses by handling ambiguities beyond person/number, and handling those with more precision. Although some of our experiments trained on random small percentages of the FST-generated data, further refinement and reduction of the data would demonstrate how the neural model performs on an incomplete selection of word forms, a situation not uncommon from hand-written descriptions of endangered languages.[5]

References

Adrian Akmajian and Stephen Anderson. 1970. On the use of fourth person in Navajo, or Navajo made harder. *International Journal of American Linguistics*, 36(1):1–8.

Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. 2015. Neural machine translation by jointly learning to align and translate. In *ICLR*.

Kenneth R. Beesley and Lauri Karttunen. 2003. *Finite State Morphology*. CSLI Publications, Stanford, CA.

Ryan Cotterell, Christo Kirov, John Sylak-Glassman, David Yarowsky, Jason Eisner, and Mans Hulden. 2016. The SIGMORPHON 2016 shared task—morphological reinflection. In *Proceedings of the 14th SIGMORPHON Workshop on Computational Research in Phonetics, Phonology, and Morphology*, pages 10–22, Berlin, Germany, August. Association for Computational Linguistics.

Ryan Cotterell, Christo Kirov, John Sylak-Glassman, Gèraldine Walther, Ekaterina Vylomova, Patrick Xia, Manaal Faruqui, Sandra Kübler, David Yarowsky, Jason Eisner, and Mans Hulden. 2017. CoNLL-SIGMORPHON 2017 shared task: Universal morphological reinflection in 52 languages. In *Proceedings of the CoNLL SIGMORPHON 2017 Shared Task: Universal Morphological Reinflection*, pages 1–30. Association for Computational Linguistics.

Andrew Cowell and Alonzo Moss Sr. 2008. *The Arapaho Language*. University Press of Colorado.

Mans Hulden. 2009. Foma: a finite-state compiler and library. In *Proceedings of the 12th Conference of the European Chapter of the Association for Computational Linguistics*, pages 29–32. Association for Computational Linguistics.

Katharina Kann, Ryan Cotterell, and Hinrich Schütze. 2017. Neural multi-source morphological reinflection. In *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume 1, Long Papers*, pages 514–524. Association for Computational Linguistics.

Lauri Karttunen. 1993. *Finite-state lexicon compiler*. Xerox Corporation. Palo Alto Research Center.

Ghazaleh Kazeminejad, Andrew Cowell, and Mans Hulden. 2017. Creating lexical resources for polysynthetic languages—the case of Arapaho. In *Proceedings of the 2nd Workshop on the Use of Computational Methods in the Study of Endangered Languages*, pages 10–18, Honolulu, March. Association for Computational Linguistics.

[5]We gratefully acknowledge the support of NVIDIA Corporation with the donation of the Titan Xp GPU used for this research.

G. Klein, Y. Kim, Y. Deng, J. Senellart, and A. M. Rush. 2017. OpenNMT: Open-Source Toolkit for Neural Machine Translation. *ArXiv e-prints*.

Kimmo Koskenniemi. 1983. *Two-level morphology: A general computational model for word-form recognition and production*. Publication 11, University of Helsinki, Department of General Linguistics, Helsinki.

Krister Lindén, Miikka Silfverberg, and Tommi Pirinen. 2009. HFST tools for morphology—an efficient open-source package for construction of morphological analyzers. In *International Workshop on Systems and Frameworks for Computational Morphology*, pages 28–47. Springer.

Peter Makarov, Tatiana Ruzsics, and Simon Clematide. 2017. Align and copy: UZH at SIGMORPHON 2017 shared task for morphological reinflection. In *Proceedings of the CoNLL SIGMORPHON 2017 Shared Task: Universal Morphological Reinflection*, pages 49–57, Vancouver, August. Association for Computational Linguistics.

Michael Maxwell. 2015. Grammar debugging. In *Systems and Frameworks for Computational Morphology*, pages 166–183. Springer.

Miikka Silfverberg and Mans Hulden. 2018. Initial experiments in data-driven morphological analysis for Finnish. In *Proceedings of the Fourth International Workshop on Computatinal Linguistics of Uralic Languages*, pages 100–107, Helsinki, Finland, January. Association for Computational Linguistics.

Ilya Sutskever, Oriol Vinyals, and Quoc V. Le. 2014. Sequence to sequence learning with neural networks. In *Advances in Neural Information Processing Systems (NIPS)*, pages 3104–3112.

Robert Young and William Morgan. 1987. *The Navajo Language: a Grammar and Colloquial Dictionary*. University of New Mexico Press, Albuquerque.

8 Appendix

Below are the results from training the neural model to produce disambiguated morphosyntactic tags both before and after a root morpheme, but evaluated only on the first set of tags and the root morpheme. The training model works with a vocabulary of 56 morphosyntactic tags and 16 letters. The 80/20/20 train/dev/test split resulted in 81,883 test examples of both ambiguous and unambiguous forms.

Tag or Letter	Precision	Recall	F_1-score	Instances
[1PL-EXCL-SUBJ.2PL-OBJ]	0.96	0.99	0.98	1381
[1PL-EXCL-SUBJ.2SG-OBJ]	1.00	0.99	0.99	1413
[1PL-EXCL-SUBJ.3PL-OBJ]	0.66	0.99	0.79	1395
[1PL-EXCL-SUBJ.3SG-OBJ]	1.00	0.50	0.66	1397
[1PL-INCL-SUBJ.3-OBJ]	0.98	1.00	0.99	2759
[1PL-INCL-SUBJ]	0.99	0.98	0.98	712
[1PL-SUBJ]	1.00	0.99	1.00	1396
[1SG-SUBJ]	0.98	0.98	0.98	673
[1SG-SUBJ][2PL-OBJ]	1.00	1.00	1.00	1407
[1SG-SUBJ][2SG-OBJ]	0.99	1.00	1.00	1350
[1SG-SUBJ][3PL-OBJ]	0.85	0.99	0.91	1363
[1SG-SUBJ][3SG-OBJ]	0.88	0.98	0.93	1384
[2PL-SUBJ.3-OBJ]	0.99	0.99	0.99	4085
[2PL-SUBJ]	0.98	0.97	0.97	1394
[2PL-SUBJ][1PL-EXCL-OBJ]	1.00	1.00	1.00	2020
[2PL-SUBJ][1SG-OBJ]	0.96	0.98	0.97	2078
[2SG-SUBJ]	1.00	0.83	0.91	1052
[2SG-SUBJ][1PL-EXCL-OBJ]	1.00	1.00	1.00	2082
[2SG-SUBJ][1SG-OBJ]	0.89	0.95	0.92	2057
[2SG-SUBJ][3PL-OBJ]	0.99	0.68	0.80	2099
[2SG-SUBJ][3SG-OBJ]	0.73	0.98	0.84	2047
[3-SUBJ.1PL-INCL-OBJ]	0.99	0.99	0.99	2751
[3-SUBJ.2PL-OBJ]	0.99	0.99	0.99	2829
[3-SUBJ.4-OBJ]	0.98	0.99	0.99	2802
[3PL-SUBJ.4-OBJ]	0.99	0.99	0.99	2767
[3PL-SUBJ]	0.99	0.85	0.91	2600
[3PL-SUBJ][1PL-EXCL-OBJ]	0.69	0.93	0.79	1352
[3PL-SUBJ][1SG-OBJ]	0.98	0.99	0.98	1351
[3PL-SUBJ][2SG-OBJ]	1.00	1.00	1.00	1384
[3SG-SUBJ]	0.98	0.81	0.89	2614
[3SG-SUBJ][1PL-EXCL-OBJ]	0.90	0.57	0.70	1344
[3SG-SUBJ][1SG-OBJ]	0.98	1.00	0.99	1298
[3SG-SUBJ][2SG-OBJ]	0.99	1.00	1.00	1401
[4-SUBJ.3PL-OBJ]	0.99	0.99	0.99	2867
[4-SUBJ.3SG-OBJ]	0.99	0.99	0.99	2788
[4-SUBJ.4-OBJ]	0.99	1.00	0.99	11016
[4PL-SUBJ]	0.95	0.98	0.97	2630
[4SG-SUBJ]	0.93	0.96	0.94	2545
[AFFIRMATIVE]	1.00	1.00	1.00	36878
[AI]	0.84	0.90	0.87	1144
[ANIMATE-OBJECT]	0.99	1.00	0.99	66267
[ANIMATE-SUBJECT]	0.84	0.90	0.87	1144
[IC]	1.00	0.99	0.99	18417

Continued on next page

Table 3 – *Continued from previous page*

Tag or Letter	Precision	Recall	F_1-score	Instances
[II]	0.98	0.93	0.95	6547
[IMPERATIVE]	0.95	0.98	0.97	3124
[INANIMATE-OBJECT]	0.99	0.92	0.95	7935
[INANIMATE-SUBJECT]	0.98	0.93	0.95	6547
[INTERROGATIVE]	1.00	1.00	1.00	19884
[NEG]	1.00	1.00	1.00	18850
[NON _ AFFIRMATIVE]	1.00	1.00	1.00	38734
[PAST]	1.00	1.00	1.00	18461
[PRESENT]	1.00	1.00	1.00	57151
[PROHIBITIVE]	1.00	1.00	1.00	3157
[TA]	0.99	1.00	0.99	66267
[TI]	0.99	0.92	0.95	7935
[VERB]	1.00	1.00	1.00	81893
b	0.95	0.99	0.97	19171
c	1.00	1.00	1.00	18440
e	0.99	0.99	0.99	85704
h	1.00	0.99	1.00	61047
i	0.99	0.99	0.99	103323
k	1.00	1.00	1.00	21331
n	0.99	0.99	0.99	71447
o	0.99	0.99	0.99	157112
s	1.00	0.99	0.99	17716
t	1.00	0.99	0.99	34310
u	0.99	1.00	0.99	38280
w	0.98	0.95	0.96	22060
x	0.99	0.99	0.99	14429
y	0.99	0.99	0.99	6978
'	1.00	1.00	1.00	35888
3	0.99	1.00	1.00	31242
average/total:	0.97	0.96	0.96	1,280,696