*INTERSPEECH 2019*
September 15–19, 2019, Graz, Austria

# Better morphology prediction for better speech systems

*Dravyansh Sharma*[1], *Melissa Wilson*[2], *Antoine Bruguier*[3]
[1]Carnegie Mellon University, USA
[2]University of Oklahoma, USA
[3]Google LLC, USA
dravyans@cs.cmu.edu, m.e.wilson@ou.edu, tonybruguier@google.com

## Abstract

Prediction of morphological forms is a well-studied problem and can lead to better speech systems either directly by rescoring models for correcting morphology, or indirectly by more accurate dialog systems with improved natural language generation and understanding. This includes both lemmatization, i.e. deriving the lemma or root word from a given surface form as well as morphological inflection, i.e. deriving surface forms from the lemma. We train and evaluate various language-agnostic end-to-end neural sequence-to-sequence models for these tasks and compare their effectiveness. We further augment our models with pronunciation information which is typically available in speech systems to further improve the accuracies of the same tasks. We present the results across both morphologically modest and rich languages to show robustness of our approach.

**Index Terms**: speech recognition, morphology, pronunciations

## 1. Introduction

Text-to-speech and speech recognition systems convert a stream of 'words' to an audio stream and vice versa. The term 'word' is often loosely used to refer to a single token in normalized text, but it is useful to distinguish between the notions of 'lexeme' and 'word-form'. A lexeme (or lemma or root) is roughly a concept which can be expressed using any of its inflected word-forms (or surface forms), e.g. the English lexeme SING has word-forms *sing*, *sings*, *sang*, *sung* and *singing*. We call these morphologically related forms. The ability to determine a word-form appropriate to the syntax of a sentence from a given lemma is critical for natural language generation and the ability to determine the lemma for a given surface form is essential for natural language understanding. Coupled with morphological tagging, we can use morphological inflectors and lemmatizers to improve the quality of natural language conversational agents. More directly, inflection is useful for text-normalization (the transformation of words from the written to the spoken form) in text-to-speech and can improve recognition quality by morphology based rescoring in speech recognition for morphologically rich languages [1]. As the example above shows, inflection can happen via different mechanisms like morpheme affixation ('-s' in sings, '-ing' in singing) or systematic vowel changes (ablaut phenomenon in sang/sung), and across languages several distinct mechanisms are known like reduplication, infixation, deletion, etc. (see Table 1)

Traditionally in computational morphology, to compute the different morphological surface forms of a lemma, language-specific hand-crafted rules and paradigms were listed and encoded as finite state transducers [2] [3] [4]. These rules are however difficult to automatically infer from a lexicon since, for example, conflict resolution when one rule acts as exception to another results in large complicated transducers. To scale the process, a number of approaches have been proposed to build models to infer the rules directly from lexical data [5] [6] [7]. Sequence to sequence deep learning approaches are promising as they avoid language-specific feature engineering, are easier to apply to new languages and yet attain state-of-the-art results [8] [9].

Table 1: *Reduplication, infixation and deletion for morphology.*

**Reduplication in Pingelapese**

| MEJR | to sleep |
|---|---|
| *mejr* | sleep |
| *mejmejr* | sleeping |
| *mejmejmejr* | still sleeping |

**Infixes and circumfixes in Tagalog**

| KAIN | to eat | KAIN | to eat |
|---|---|---|---|
| *kumain* | eat! (imperative) | *pakainin* | to let eat |
| *kinain* | ate (past) | *pagkainan* | eat in/at |

**Subtractive morphology in Mikasuki**

| HOFAALI | to take/pull out |
|---|---|
| *hofaali* | to take out (sg obj) |
| *hofli* | to take out (pl obj) |

In this work we present a detailed evaluative study of different language-independent neural paradigms applied to the task of morphological inflection as well as lemmatization, where we frame both problems as sequence to sequence string transductions with the morphology class as an additional input. In addition to unifying the two symmetric problems by approaches that are effective for both, our approaches involve generic solutions for injecting a non-sequential feature (the morphology class) into the various well-known sequence transduction neural paradigms.

We further extend the work to improve the models by adding pronunciation data (word to phoneme mappings) for both tasks, which is typically readily available in speech systems. The linguistic reason why this helps is because often morphological changes are a function of phonemic information which may not be transparent in orthography. This requires extending the above approaches to inject non-sequential data to approaches that can inject sequential auxiliary data. The approaches are fully general and can be applied in a wide variety of settings to add features to sequence-to-sequence neural paradigms.

Morphology is also used to refer to the process of word formation by derivation or compounding. In this work we focus our attention on inflectional morphology although it would be interesting to study derivation and compounding by appropriately adapting the techniques discussed here. Improving compounding can lead to improvements in speech recognition systems of languages with productive compounding like German [10].



http://dx.doi.org/10.21437/Interspeech.2019-3207

Table 2: *Sample input output alignment for character LSTM model.*

| **Input** | <B> | S | I | N | G | <E> | $\phi$ | $\phi$ | $\phi$ | $\phi$ | $\phi$ | $\phi$ | $\phi$ | $\phi$ | $\phi$ | $\phi$ | $\phi$ | $\phi$ |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **Output** | $\phi$ | $\phi$ | $\phi$ | $\phi$ | $\phi$ | <B> | s | a | n | g | <E> | $\phi$ | $\phi$ | $\phi$ | $\phi$ | $\phi$ | $\phi$ | $\phi$ |

# 2. Related Work

## 2.1. Morphology and speech

Morphology prediction is a problem of independent interest and has applications in natural language generation as well as natural language understanding. The problems of lemmatization and morphological inflection have been studied in both contextual (in a sentence, which involves morphosyntactics) and isolated settings [8] [11] [12].

Speech recognition and synthesis systems can be viewed as sequence-to-sequence transduction problems and various end-to-end models have gathered recent interest [13] [14]. End-to-end systems are an attractive option for small sized models which can potentially take the complete context of the problem into consideration, but most state-of-the-art conversational systems are a composition of several sub-systems, often each being a sequence-to-sequence problem. Morphology prediction finds applications in several positions in this stack of sub-systems, and can also be used as a reranking tool to improve accuracy in end-to-end recognition systems [1] and prosody in synthesis systems [15].

## 2.2. Neural networks for sequence learning

Recurrent Neural Networks (RNNs) are able to learn how to map a given input sequence to an output sequence of the same length and can 'remember' previous inputs/outputs by reusing their outputs as an additional input. However vanilla RNNs tend to quickly forget their past inputs due to a problem known as vanishing gradients. A popular resolution is to use LSTMs to allow remembering relevant information for arbitrary time lengths [16]. Further bi-directional LSTMs allow initial output to depend on input symbols that appear later in the sequence.

RNNTs (Recurrent neural network transducer) constitute an effective sequence-to-sequence learning paradigm in the encoder-decoder paradigm [17] which allows transduction of variable length sequences. The transducers also allow for an explicit mapping between input and output symbol subsequences. The dependence on far-away elements is possible due to the RNN units.

Recently attention-based sequence to sequence models have become popular [18] [19] [20] [21] as they allow a more efficient way of focussing on just the relevant sections of the input to produce the corresponding output segment.

Given the variety of sequence learning neural architectures available, studies comparing their effectiveness for specific application paradigms are useful [22] [23].

# 3. Sequence-to-Sequence Models

For concreteness, we focus on the joint supervised setting i.e. we use annotated data to learn a single model for predicting inflections/lemmata for all morphological classes. The joint setting is the most relevant for scaling to low-resource languages as it allows sharing of learnt information across inflectional paradigms, and typically the most effective models in state-of-the-art approaches are built in this setting.

We train and evaluate our models for three languages: English (United States), German (Germany) and Russian (Russia). For training our models, we use pronunciation lexicons (word-pronunciation pairs) and morphological lexicons (word-lemma-morphology feature vectors) of size $\approx 10^5$-$10^6$ for each language. For the languages discussed, such lexicons can be obtained for example by scraping Wiktionary data. While this is typical high-resource language domain, in section 6 we discuss applications and extension of our work to low-resource settings. We keep $20\%$ of the lexicons aside for evaluation using word error rate metric.

Morphological classes are often described as a complex hierarchy, e.g. tense-mood-aspect are features typically applicable only for the verb part-of-speech, but the surface form may also be influenced by number and person. For simplicity, we treat all features independently, with an additional value '$\langle$N/A$\rangle$' (not applicable) if a particular feature does not apply to a word. Thus morphology class is encoded as a fixed length vector for each word, for example

$$\begin{array}{rl} & \text{P.O.S. degree number tense} \ldots \\ \text{M(happier)} = & \langle adj. \rangle \, \langle comp. \rangle \, \langle N/A \rangle \, \langle N/A \rangle \ldots \end{array}$$

Morphology class is thus an integer sequence if each feature value is mapped to a distinct integer in the range $[1, \#(\text{legal values for feature})+1]$. Note that the same spelling can appear multiple times in the morphology lexicon corresponding to different words with potentially different pronunciations. In all our models we allow for missing/incomplete data by using a special '$\langle$UNKNOWN$\rangle$' symbol which the models learn to ignore.

In the following subsections we describe common algorithms for lemmatization and inflection (i.e. training with input and output switched) and compare baseline accuracies for sequence-to-sequence transduction with those of models augmented with morphology class information.

## 3.1. Character LSTM (long short-term memory) Models

The input and output orthographic sequences (encoded as one-hot vectors) are padded to be of the same length (we use an upper bound on the total input+output length for the language), and the output is translated to coincide with the end of input. For example for the '`SING`, sang' pair, the input and output encoding is shown in Table 2. This allows the model to see the entire input sequence first and then start predicting the output sequence.

Input and output tapes are switched for the lemmatization problem. To augment with morphology class in this model, we simply extend the input tape with a section encoding the morphology feature vector for the surface form. For lemmatization we build two models, a baseline without injecting morphology class (which corresponds to lemmatization in absence of morphological tagging, 'LSTM-base' in Table 3) and one with the morphology class added to input ('LSTM' in Table 3).

The accuracy of prediction for the inflection task is noted in Table 4.

Table 3: *Lemmatization accuracy using various sequence-to-sequence neural models. 'base' here refers to lemmatization without knowledge of morphology class*

| Model | *English* | *German* | *Russian* |
|---|---|---|---|
| **LSTM-base** | 85.5% | 69.0% | 82.0% |
| **LSTM** | 94.4% | 85.9% | 89.1% |
| **RNNT-base** | 95.5% | 73.2% | 88.8% |
| **RNNT** | **97.8**% | 91.9% | 93.1% |
| **Attention-base** | 94.6% | 81.1% | 89.0% |
| **Attention** | 97.5% | **93.2**% | **94.5**% |

Table 4: *Inflection accuracy using various sequence-to-sequence neural models.*

| Model | *English* | *German* | *Russian* |
|---|---|---|---|
| **LSTM** | 96.1% | 93.7% | 63.6% |
| **RNNT** | 98.1% | 95.7% | 75.2% |
| **Attention** | **98.7**% | **95.9**% | **79.4**% |

### 3.2. RNN Transducers

Recurrent neural network transducers have a jointly trained encoder-decoder architecture. We use bidirectional RNN layers with dropout to avoid overfitting. The morphology feature is tiled along the input to ensure it is considered throughout the transduction (i.e. appended to each input/output symbol instead of just at the end). We also tried injecting the morphology feature in the decoding/joint layers but there was negligible change relative to embedding in just the encoding layer.

The accuracy of prediction for the lemmatization and inflection tasks are noted in Table 3 and Table 4 respectively.

### 3.3. Attention-based Models

We also study effectiveness of dot-product (soft) attention models for lemmatization and inflection. Here, instead of the joint network trained in RNNTs, an attention network is trained to learn a weighted linear combination of the entire input at each time step to learn which segment(s) of the input to focus on for a particular output segment. We simply concatenate the morphology feature vector to the input and let the attention mechanism learn the appropriate segments of input for performing the morphological transformations. We essentially use the same input/output encoding scheme as before except that we dont need padding.

The accuracy of prediction for the lemmatization and inflection tasks are noted in Table 3 and Table 4 respectively. Attention-based models seem to perform the best accross languages for either task.

## 4. Augmenting with pronunciation data

Pronunciation data is available as pronunciation lexicons. For the morphological inflection and lemmatization tasks we can lookup the pronunciation (i.e. a phoneme sequence with stress symbols and syllable boundaries marked) from the lexicon during inference. So it makes sense to augment the input sequence (lemma or surface form depending on the task) with its pronunciation during training and see if we get any improvements. This is expected to help in case of homographs (words with same spelling but different pronunciations e.g. 'resent' - if the pronunciation is /ɹɪˈzɛnt/ it corresponds to the lemma 'RESENT', but /ˌiːˈsɛnt/ is the past form of 'RESEND') but also more generally because morphological changes are often a function of phonology. For instance, 'thief' and 'life' in English undergo a similar transformation to give plurals 'thieves' and 'lives'. Although the surface forms end differently, the pronunciations end in the same phoneme /f/ which dictates the peculiar pluralization. Thus we hope to resolve these kinds of errors which the models described in the previous section are likely to make, by developing pronunciation informed models.

Table 5: *Lemmatization accuracy using various sequence-to-sequence neural models with input augmented with pronunciation.*

| Model | *English* | *German* | *Russian* |
|---|---|---|---|
| **LSTM** | 95.4% | 87.1% | 89.5% |
| **RNNT** | 97.9% | 92.2% | 93.1% |
| **Attention** | **98.0**% | **93.9**% | **94.7**% |

Table 6: *Inflection accuracy using various sequence-to-sequence neural models, with input augmented with pronunciation.*

| Model | *English* | *German* | *Russian* |
|---|---|---|---|
| **LSTM** | 97.1% | 94.9% | 65.5% |
| **RNNT** | 98.5% | 95.7% | 75.5% |
| **Attention** | **98.8**% | **96.3**% | **81.2**% |

Pronunciation however, unlike morphology class, is sequential data. Hence, to use it effectively, the model must also learn to align the pronunciation to the input word/lexeme. Depending on the neural architecture we can have different ways to inject the pronunciation into the sequence-to-sequence paradigm.

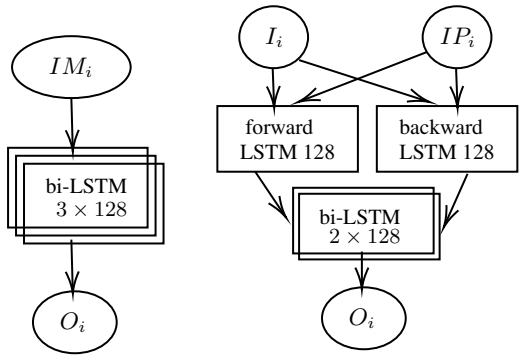


Figure 1: *LSTM architectures for injecting pronunciation for morphology prediction. $IM$ is input word/lexeme sequence $I$ augmented by morphology class $M$, $O$ is the output word/lexeme and $IP$ is the input pronunciation.*

### 4.1. Character LSTM Models

Lemma pronuciation can be added to LSTM models by adding a symmetric additional input (paddings adjusted to make all words and pronunciations fit). See Figure 1 to compare the

archictecture here with that in section 3.1.

### 4.2. RNN Transducers

For RNNTs, we can first align the input word/lexeme with its pronunciation, and finally align the output with the aligned graphone sequence. The first alignment results in a transducer, transduction steps of which (correspond to advancing along the input) are mapped to output sequence. So during inference we simply apply the transduction and pick the most likely output with beam search.

### 4.3. Attention-based Models

In attention-based models we can simply concatenate the two pronunciations in the input (punctuated by a suitable boundary symbol) since the model can simultaneously look at distant portions of the input sequence efficiently. Another alternative is 2-D attention as described in [24].

The accuracies for these various approaches described above are noted in Table 4 and 5 for the lemmatization and inflection task respectively. Again we note attention-based models perform well accross tasks and languages.

Note that lemmatization is unique for a word form and is easier to do without morphology information of the source form. Thus lemmatization baselines are typically high even without the morphology class information and augmenting with additional data does not improve it as well as the inflection task.

## 5. Conclusion

We note that common sequence-to-sequence neural models can be readily extended by injecting morphology and produce state-of-the-art accuracies on lemmatization and inflection tasks, and are significantly better than vanilla LSTMs which don't capture the transduction or injection as effectively. Addition of non-sequential and sequential auxiliary data to well-known sequence-to-sequence neural architectures is explored and we have successfully demonstrated how these can be useful in improving system accuracy on lemmatization and inflection tasks, by augmenting the input to the models with morphology class and pronunciation data.

## 6. Discussion and future work

One way to extend the work here is to exploit the hierarchical nature of morphology classes, possibly by encoding using more powerful models like structured attention [25] [21]. Partial analysis of morphology can be used - often full analysis is either not available or in fact not unambiguous given the context.

Also it would be interesting to extend the work to derivation and compounding. Extension to derivation is rather simple, but compounding (analysing compounds just given their spelling and morphology class) is hard and would likely need memory augmented models [26]. Another venue to explore is extension to contextual settings, one could simultaneously compute morphosyntactic tagging and desired morphological form or in certain applications skip morphosyntactic tagging altogether [27]. Linguistically rich morphology allows the language to have a looser syntax and word ordering requirements, so it would be interesting to see a unified mechanism to generate or analyse both simultaneously [28].

The approaches described in this work are fully supervised. It would be interesting to see more cross-lingual and semi-supervised solutions to these problems to extend them better to the low-resource languages. On the other hand, models like these can help development of lexical resources more readily in low resource settings - one could largely automate generation of morphological forms and speed up lexical development by requiring only annotations of root forms [29].

## 7. Acknowledgments

We thank our colleagues Markus Becker, Ivan Korotkov, Leif Johnson and Neha Chaudhari for helpful discussions. The work was done while the authors were at Google.

## 8. References

[1] I. Shafran and K. Hall, "Corrective models for speech recognition of inflected languages," in *Proceedings of the 2006 Conference on Empirical Methods in Natural Language Processing*. Association for Computational Linguistics, 2006, pp. 390–398.

[2] K. Koskenniemi, "Two-level model for morphological analysis." in *IJCAI*, vol. 83, 1983, pp. 683–685.

[3] R. W. Sproat *et al.*, *Morphology and computation*. MIT press, 1992.

[4] R. M. Kaplan and M. Kay, "Regular models of phonological rule systems," *Computational linguistics*, vol. 20, no. 3, pp. 331–378, 1994.

[5] G. Chrupała, "Simple data-driven context-sensitive lemmatization," *Procesamiento del lenguaje natural, nº 37 (sept. 2006), pp. 121-127*, 2006.

[6] R. Roth, O. Rambow, N. Habash, M. Diab, and C. Rudin, "Arabic morphological tagging, diacritization, and lemmatization using lexeme models and feature ranking," *Proceedings of ACL-08: HLT, Short Papers*, pp. 117–120, 2008.

[7] G. Nicolai, C. Cherry, and G. Kondrak, "Inflection generation as discriminative string transduction," in *Proceedings of the 2015 conference of the North American chapter of the association for computational linguistics: human language technologies*, 2015, pp. 922–931.

[8] M. Faruqui, Y. Tsvetkov, G. Neubig, and C. Dyer, "Morphological inflection generation using character sequence to sequence learning," *arXiv preprint arXiv:1512.06110*, 2015.

[9] R. Aharoni and Y. Goldberg, "Morphological inflection generation with hard monotonic attention," *arXiv preprint arXiv:1611.01487*, 2016.

[10] H. Lüngen, M. Pampel, G. Drexel, D. Gibbon, F. Althoff, and C. Schillo, "Morphology and speech technology," 1996.

[11] S. B. Cohen and N. A. Smith, "Joint morphological and syntactic disambiguation," in *Proceedings of the 2007 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning (EMNLP-CoNLL)*, 2007.

[12] R. Cotterell, C. Kirov, J. Sylak-Glassman, D. Yarowsky, J. Eisner, and M. Hulden, "The sigmorphon 2016 shared taskmorphological reinflection," in *Proceedings of the 14th SIGMORPHON Workshop on Computational Research in Phonetics, Phonology, and Morphology*, 2016, pp. 10–22.

[13] Y. Miao, M. Gowayyed, and F. Metze, "Eesen: End-to-end speech recognition using deep rnn models and wfst-based decoding," in *2015 IEEE Workshop on Automatic Speech Recognition and Understanding (ASRU)*. IEEE, 2015, pp. 167–174.

[14] Y. Wang, R. Skerry-Ryan, D. Stanton, Y. Wu, R. J. Weiss, N. Jaitly, Z. Yang, Y. Xiao, Z. Chen, S. Bengio *et al.*, "Tacotron: Towards end-to-end speech synthesis," *arXiv preprint arXiv:1703.10135*, 2017.

[15] M. Vainio *et al.*, "Artificial neural network based prosody models for finnish text-to-speech synthesis," 2001.

[16] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural computation*, vol. 9, no. 8, pp. 1735–1780, 1997.

[17] A. Graves, "Sequence transduction with recurrent neural networks," *arXiv preprint arXiv:1211.3711*, 2012.

[18] W. Chan, N. Jaitly, Q. Le, and O. Vinyals, "Listen, attend and spell: A neural network for large vocabulary conversational speech recognition," in *Acoustics, Speech and Signal Processing (ICASSP), 2016 IEEE International Conference on*. IEEE, 2016, pp. 4960–4964.

[19] M.-T. Luong, H. Pham, and C. D. Manning, "Effective approaches to attention-based neural machine translation," *arXiv preprint arXiv:1508.04025*, 2015.

[20] K. Xu, J. Ba, R. Kiros, K. Cho, A. Courville, R. Salakhudinov, R. Zemel, and Y. Bengio, "Show, attend and tell: Neural image caption generation with visual attention," in *International conference on machine learning*, 2015, pp. 2048–2057.

[21] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin, "Attention is all you need," in *Advances in neural information processing systems*, 2017, pp. 5998–6008.

[22] R. Prabhavalkar, K. Rao, T. N. Sainath, B. Li, L. Johnson, and N. Jaitly, "A comparison of sequence-to-sequence models for speech recognition." in *Interspeech*, 2017, pp. 939–943.

[23] S. Bai, J. Z. Kolter, and V. Koltun, "An empirical evaluation of generic convolutional and recurrent networks for sequence modeling," *arXiv preprint arXiv:1803.01271*, 2018.

[24] A. Bruguier, H. Zen, and A. Arkhangorodsky, "Sequence-to-sequence neural network model with 2d attention for learning japanese pitch accents," *Proc. Interspeech 2018*, pp. 1284–1287, 2018.

[25] Y. Kim, C. Denton, L. Hoang, and A. M. Rush, "Structured attention networks," *arXiv preprint arXiv:1702.00887*, 2017.

[26] A. Bruguier, A. Bakhtin, and D. Sharma, "Dictionary augmented sequence-to-sequence neural network for grapheme to phoneme prediction," *Proc. Interspeech 2018*, pp. 3733–3737, 2018.

[27] T. Müller, R. Cotterell, A. Fraser, and H. Schütze, "Joint lemmatization and morphological tagging with lemming," in *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, 2015, pp. 2268–2274.

[28] M. Korobov, "Morphological analyzer and generator for russian and ukrainian languages," in *International Conference on Analysis of Images, Social Networks and Texts*. Springer, 2015, pp. 320–332.

[29] M. Faruqui, R. McDonald, and R. Soricut, "Morpho-syntactic lexicon generation using graph-based semi-supervised learning," *Transactions of the Association for Computational Linguistics*, vol. 4, pp. 1–16, 2016.