Experiments on Morphological Reinflection: CoNLL 2018 Shared Task

Rishabh Jain
IIT (BHU), Varanasi, India
rishabh.j.017@gmail.com

Anil Kumar Singh
IIT (BHU), Varanasi, India
nlprnd@gmail.com

Abstract

We present a system for the task of morphological inflection, i.e., finding a target morphological form, given a lemma and a set of target tags. System is trained on datasets of three sizes: low, medium and high. The system uses a simple Long Short-Term Memory (LSTM) based encoder-decoder based model. The performance for low size dataset is poor in general while it improves significantly for medium and high sized training dataset. The average performance over all languages is poor as compared to baseline for low dataset, it is comparable for medium dataset, and significantly more for high dataset.

1 Introduction

The CoNLL-SIGMOPRHON 2018 shared task consists of two subtasks out of which we participate only in the first subtask, which involves generating a target inflected form from a given lemma with its morphosyntactic descriptions (MSDs) provided as a set of features. For instance, the word thinking is the present continuous inflected form of the lemma think. The models were trained on three differently-sized datasets. The low-sized datasets had around 100 training samples, the medium-sized datasets had around 1000 training samples and the high-sized datasets had around 10000 samples for most languages. Datasets were provided for a total of 103 languages including surprise data.

2 Background

Prior to neural network based approaches to morphological reinflection, most systems used a 3-step approach to solve the problem:
1) String alignment between the lemma and the target (morphologically transformed form),
2) Rule extraction from spans of the aligned strings and
3) Rule application to previously unseen lemmas to transform them.
(Durrett and DeNero, 2013) and (Ahlberg et al., 2014, 2015) used the above approaches, with each of them using different string alignment algorithms and different models to extract rules from these alignment tables. However, in these kinds of systems, the types of rules to be generated must be specified, which should also be engineered to take into account language-specific transformational behaviour.

(Faruqui et al., 2016) proposed a neural network based system which abstracts away the above steps by modeling the problem as one of generating a character sequence, character-by-character. (Kann and Schütze, 2016) proposed a highly competetive implementation in previous year tasks (Cotterell et al., 2016, 2017).

Akin to machine translation systems, this system uses an encoder-decoder LSTM model as proposed by (Hochreiter and Schmidhuber, 1997). The encoder is a bidirectional LSTM, while the decoder LSTM feeds into a softmax layer for every character position in the target string. Decoder predicts the output sequence character by character using feedback until stop is predicted. This model takes into account the fact that the target and the root word are similar, except for the parts that have been changed due to inflection, by feeding the root word directly to the decoder as well. A separate neural net is trained for every language.

3 System Description

We have modelled our system based on the system proposed by (Faruqui et al., 2016), as described in the previous section. However we have made some modifications to the above system, to account for the three different sizes of datasets and to

Proceedings of the CoNLL–SIGMORPHON 2018 Shared Task: Universal Morphological Reinflection, pages 48–57,
Brussels, Belgium, October 31, 2018.

account for the behaviour of morphological transformations of independent languages.

In the model, some structural and hyper-parameter features remain the same. The characters in the root word and morphological features of the tatrget word are represented using one hot vectors. The major change in our model is the size of LSTM layers which is kept variable (depending on vocabulary size) as opposed to fixed as in system proposed by (Faruqui et al., 2016) based on assumption that bigger vocabulary would require bigger layers to extract features and system is trained for more epochs.

The embedding size for each language is different depending upon the alphabet set of that language available in the given dataset and similarly for morphological tags which are split into individual components. We use a bidirectional encoder to which we feed the input word embeddings. The output of the encoder, concatenated with the root word embedding and morphological features, feeds into the decoder. All recurrent units have variable hidden layer dimensions depending upon the embedding size of root word and morphological features. Over the decoder layer is a softmax layer that is used to predict the character that must occur at each character position of the target word. In order to maintain a constant word length, we use paddings of 0 characters. All models use categorical cross-entropy as the loss function and the RMSProp optimizer for optimization.

The model was trained for 100 epochs for each size. Keras API (Chollet et al., 2015) was used for writing neural networks. For low dataset, batch size of 10 was used, for medium 100, and for high 250/500 depending of hardware limitations.

Submission

Following are tables showing top 5 accuracies obtained by our system on test data as opposed to baseline model.

3.1 Low-sized Dataset

Language	Baseline	Enc-Dec
Telugu	70	94
Uzbek	52	82
Karelian	24	80
Mapudungun	64	74
Kazakh	26	68

Table 1: Top 5 Accuracies for languages for low data

3.2 Medium-sized Dataset

Language	Baseline	Enc-Dec
Uzbek	96	100
Classical-syriac	99	99
Crimean-tatar	78	98
Khakas	84	98
Mapudungun	82	98

Table 2: Top 5 Accuracies for languages for medium data

3.3 High-sized Dataset

Language	Baseline	Enc-Dec
Classical-syriac	97	100
Crimean-tatar	95	100
Haida	66	100
Swahili	N/A	100
Kannada	66	100

Table 3: Top 5 Accuracies for languages for high data

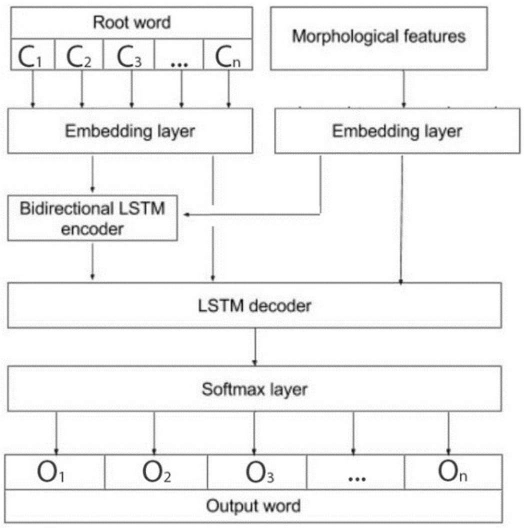

Figure 1: C1, .., Cn represent characters of the root word while O1, ..,On represent characters of the output word.

4 Evaluation

4.1 Results on Test Set

The evaluation results were obtained using the evaluation script and the test set provided by the shared task organizers.

The best five baseline accuracies, accuracies for the first submission and accuracies for the second submission can be found in Table 1, Table 2 and Table 3 for each of the three dataset sizes: low, medium and high respectively.

The complete set of accuracies and Levenshtein distances for all languages have been included in Appendix (tables 4 to 6).

4.2 Observations

We performed some experiments, where the choice of hyperparameters was guided by intuitions developed from analysis of the dataset and results obtained on smaller subsets of the data. We have presented some key observations from our analysis in the ensuing sub-sections.

4.2.1 Number of layers

We observed that increasing number of layers does not result in significant increase in performance, even reduced performance in some cases whereas increased computation time significantly. So instead of adding more layers, adding more complexity and features in current layer is bound to improve performance.

4.2.2 Embedding of Morphological features

Multiple types of embedding to represent morphological features was tried some of which were: binary vectors, one hot vectors, integer vectors. One hot vectors resulted in best performance for our model.

4.2.3 Size of encoder layer

Increasing size of encoder after certain multiple of total embedding size ($\sim$5) results in saturation of performance.

4.2.4 Hyperparameter Optimization

Various hyperparameters need to be optimized such as batch-size, dropout rate, number of epochs etc. which may be different for each language, to obtain optimal performance.

5 Conclusions

There are two main conclusions. One is that different configurations of deep neural networks work well for different languages. The second is that deep learning may not be the right approach for low-sized data or some other pre-processing and post-processing may need to be done to increase performance. Data augmentation is one alternative to deal with low resource languages.
Results for low-size were poor for almost all languages. So, deep learning cannot extract features adequately from low resources without data augmentation. It is to be noted that we used purely deep learning. If deep learning is augmented with other transduction, rule-based or knowledge-based methods, the results for low-size could perhaps be improved. Very high accuracies (>95%) are observed for some languages in high sized datasets, neural networks is probably the best choice for processing such languages.

References

Malin Ahlberg, Markus Forsberg, and Mans Hulden. 2014. Semi-supervised learning of morphological paradigms and lexicons. In *Proceedings of the 14th Conference of the European Chapter of the Association for Computational Linguistics: Language Technology (Computational Linguistics)*, pages 569–578.

Malin Ahlberg, Markus Forsberg, and Mans Hulden. 2015. Paradigm classification in supervised learning of morphology. In *Proceedings of the 2015 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 1024–1029.

François Chollet et al. 2015. Keras. https://keras.io.

Ryan Cotterell, Christo Kirov, John Walther, Sylak-Glassman Géraldine, Ekaterina Vylomova, Patrick Xia, Manaal Faruqui, Sandra Kübler, David Yarowsky, Jason Eisner, and Mans Hulden. 2017. The conllsigmorphon 2017 shared task: Universal morphological reinflection in 52 languages. In *Proceedings of the CoNLL-SIGMORPHON 2017 Shared Task: Universal Morphological Reinflection*.

Ryan Cotterell, Christo Kirov, John Walther, Sylak-Glassman Géraldine, Ekaterina Vylomova, Patrick Xia, Manaal Faruqui, David Yarowsky, Jason Eisner, and Mans Hulden. 2016. The sigmorphon 2016 shared taskmorphological reinflection. In *Proceedings of the 14th SIGMORPHON Workshop on Computational Research in Phonetics, Phonology, and Morphology*. ACL.

Greg Durrett and John DeNero. 2013. Supervised learning of complete morphological paradigms. In *Proceedings of the 2013 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 1185–1195.

Manaal Faruqui, Yulia Tsvetkov, Graham Neubig, and Chris Dyer. 2016. Morphological inflection generation using character sequence to sequence learning. In *Proceedings of NAACL*.

Sepp Hochreiter and Jürgen Schmidhuber. 1997. Long short-term memory. *Neural Comput.*, 9(8):1735–1780.

Katharina Kann and Hinrich Schütze. 2016. Med: The lmu system for the sigmorphon 2016 shared task on morphological reinflection. In *Proceedings of the 14th SIGMORPHON Workshop on Computational Research in Phonetics, Phonology, and Morphology*, pages 62–70.

A Appendix

In Tables 4 to 6 (on this page and the next), BA stands for baseline accuracy, L.D. for Levenshtein Distance, Acc for Accuracy, dev for development data.

Table 4: Results for all languages for low data.

languages	BA	Acc - dev	Avg. L.D. -dev	Acc -test
adyghe	59	56.9	0.908	57.8
albanian	22.3	0.9	6.736	0.8
arabic	25.6	0.5	5.489	0
armenian	37	5	4.575	5.8
asturian	58.6	24	1.882	23.9
azeri	24	22	2.3	28
bashkir	39.4	37.5	1.468	35.8
basque	0.1	1	4.949	0.5
belarusian	6.8	4.9	4.062	2.9
bengali	50	22	2.43	26
breton	20	36	1.49	30
bulgarian	30.7	9.3	3.162	9.7
catalan	60.8	20.7	1.867	25.7
classical-syriac	94	63	0.54	62
cornish	10	28	1.9	32
crimean-tatar	56	48	0.81	51
czech	38.5	7.3	3.294	8.1
danish	58.3	42.1	1.159	40.9
estonian	21.5	2.3	4.851	2.6
faroese	34.4	10.5	2.915	9.6
finnish	17.3	0.4	7.573	0.6
friulian	70	38	1.37	39
galician	53	17.3	2.349	15.3
georgian	70.6	23.1	2.087	23.8
greek	25.3	2.8	4.75	2.4
greenlandic	50	54	0.72	60
haida	29	20	4.66	12
hebrew	24.4	4.8	2.556	5.2
hindi	31.8	23.5	2.603	23.2
hungarian	17.4	4.3	3.166	4.7
icelandic	35.6	8	2.647	8.5
ingrian	20	18	2.14	24
irish	30.3	1.6	7.413	1.3
italian	40.5	10.7	3.916	11
kabardian	72	54	1.13	56
karelian	24	60	0.58	80
kashubian	60	46	0.9	46
kazakh	26	78	0.3	68
khakas	26	72	0.44	62
khaling	3.1	2.7	3.872	2.2
kurmanji	82.7	25.6	1.927	25.4
ladin	58	32	1.41	33
latin	16	0.5	4.697	1
latvian	52.2	8.1	2.757	9
lithuanian	23.3	2.5	3.789	2.8

livonian	28	6	3.51	6
lower-sorbian	32.1	7.1	2.705	9.9
macedonian	49.8	16.1	1.884	16.7
maltese	9	4	2.73	8
mapudungun	64	82	0.3	74
middle-french	76.9	33.3	1.728	32.8
middle-high-german	38	60	0.86	56
murrinhpatha	2	26	1.98	28
navajo	0	0.6	5.847	0.9
neapolitan	79	46	1.27	45
norman	30	62	0.9	52
northern-sami	16.4	2.2	4.433	1.5
norwegian-bokmaal	67.8	43.7	1.012	40.8
norwegian-nynorsk	48.9	22.4	1.749	22
occitan	72	43	1.5	31
old-armenian	31	2	3.7	1.9
old-church-slavonic	39	24	1.95	21
old-french	32.5	6	3.315	7
old-irish	8	4	3.68	2
old-saxon	22.8	3.7	3.016	5.6
pashto	35	7	2.64	11
persian	26.3	6.4	4.253	6.3
portuguese	62.6	20	1.981	19.9
quechua	15.9	15.9	3.963	15
romanian	44.8	4.3	4.352	4.6
sanskrit	33.7	6	3.339	6.7
scottish-gaelic	46	38	2.02	48
serbo-croatian	21.7	3.9	4.542	4.1
slovak	37.7	9.8	2.241	9.6
slovene	32.3	12.2	2.064	0.4
sorani	19.3	2.5	4.466	1.3
spanish	61.8	12.2	3.076	13
swahili	0	6	3.06	4
swedish	51.1	33.2	1.383	32.8
tatar	52	40	1.07	44
telugu	70	100	0	94
tibetan	34	36	1.2	36
turkish	13.2	9.4	4.398	10.7
turkmen	34	62	0.68	64
ukrainian	38.7	8.6	2.485	9
urdu	32.7	30.9	2.203	32.6
uzbek	52	83	0.36	82
venetian	71.8	29	1.457	31.1
votic	17	20	1.92	20
welsh	30	9	3.04	11
west-frisian	50	22	2.14	26
yiddish	78	29	2.15	33
zulu	0.1	1.3	5.114	1.1
dutch	50.8	9	2.665	8.9
english	77.6	58.9	0.78	61.8
french	59	11.6	2.772	13.4

german	49.2	19.6	1.991	19.9
kannada	33	17	3.65	29
middle-low-german	18	24	2.5	14
north-frisian	31	14	3.65	18
old-english	17.6	7.1	2.879	8.6
polish	40.4	4.5	3.403	4.7
russian	43.4	6.5	3.398	7.5

Table 5: Results for all languages for medium data.

languages	BA	Acc - dev	Avg. L.D. -dev	Acc -test
adyghe	84.8	94.2	0.11	92.9
albanian	61.8	39.2	1.901	39.2
arabic	39.5	39.4	2.108	1.6
armenian	70.4	62.8	1.085	67.4
asturian	89.1	87.9	0.26	90
azeri	50	89	0.35	94
bashkir	72.6	94.3	0.118	95.4
basque	1.9	66.7	0.83	69.2
belarusian	21.5	45.9	1.997	45.8
bengali	76	94	0.17	97
breton	67	86	0.37	94
bulgarian	70.8	36.6	1.524	35
catalan	85.6	83.2	0.391	83.8
classical-syriac	99	97	0.03	99
cornish	12	64	0.58	66
crimean-tatar	78	94	0.06	98
czech	79.9	61	1.07	61.1
danish	77.8	76.3	0.478	75.7
estonian	62.9	49.8	1.442	46.5
faroese	65.2	47.1	1.159	48.9
finnish	44.1	19.8	3.105	22
friulian	92	92	0.15	92
galician	82.8	81.3	0.385	82.5
georgian	92.1	87.8	0.348	91.2
greek	59.3	33.8	2.15	29.9
greenlandic	72	66	0.54	84
haida	61	93	0.16	90
hebrew	38.1	61.9	0.714	64.7
hindi	86.5	88.2	0.315	87.7
hungarian	44.4	56.7	0.89	57.1
icelandic	58.9	47	1.101	45.7
ingrian	46	84	0.4	88
irish	44	20.7	3.73	18.8
italian	72.5	67.1	0.948	68.9
kabardian	83	99	0.01	97
karelian	42	96	0.08	96
kashubian	68	80	0.22	84
kazakh	50	62	0.46	50

khakas	84	94	0.1	98
khaling	17.9	60.6	0.956	59.4
kurmanji	85.2	85.4	0.401	86.1
ladin	86	91	0.15	91
latin	37.6	28.3	1.56	31.3
latvian	85.5	73.7	0.591	73.7
lithuanian	52.2	44.9	1.216	46.8
livonian	51	60	0.84	62
lower-sorbian	68.9	65.7	0.644	69.8
macedonian	82.6	77.6	0.408	75.1
maltese	20	90	0.17	85
mapudungun	82	100	0	98
middle-french	90.3	91.2	0.242	89.4
middle-high-german	54	98	0.08	92
murrinhpatha	0	96	0.08	96
navajo	0	16.4	3.227	19.5
neapolitan	94	98	0.02	98
norman	46	20	2.18	26
northern-sami	34.8	41.6	1.554	39.4
norwegian-bokmaal	80.7	80.5	0.321	81.5
norwegian-nynorsk	61.1	56	0.746	57.5
occitan	92	94	0.15	93
old-armenian	67.3	53.7	1.168	57.3
old-church-slavonic	76	83	0.32	87
old-french	63.1	61.5	0.989	60.6
old-irish	16	30	1.7	24
old-saxon	39	63.2	0.715	64.7
pashto	69	70	0.68	73
persian	65.7	70.1	0.899	70.9
portuguese	92.4	33.5	1.858	37.3
quechua	70.9	76.7	0.693	77.3
romanian	69.4	48	1.752	48.5
sanskrit	59.7	65.9	0.645	68.8
scottish-gaelic	50	80	0.5	86
serbo-croatian	68.2	39.1	1.866	43.4
slovak	71.1	58	0.76	58.7
slovene	72.3	68	0.575	9.9
sorani	51.7	52.4	1.22	55.9
spanish	86.3	76.6	0.744	75.7
swahili	0	87	0.36	87
swedish	76.5	32	1.572	31.4
tatar	89	93	0.07	96
tibetan	36	30	1.42	22
turkish	32.8	78.9	0.661	79.2
turkmen	68	92	0.16	96
ukrainian	74.1	44.6	1.041	46.6
urdu	87.6	88.7	0.306	87.5
uzbek	96	100	0	100
venetian	89.1	92.3	0.116	91.9
votic	34	81	0.32	78
welsh	58	85	0.32	81

west-frisian	65	95	0.12	94
yiddish	87	83	0.47	88
zulu	0.1	57	1.387	52.6
dutch	72.4	69.4	0.644	71
english	90.5	90.1	0.205	90.9
french	73.2	68.2	0.881	69.3
german	71.7	69.7	0.827	65.5
kannada	55	92	0.21	85
middle-low-german	38	94	0.06	92
north-frisian	33	85	0.36	85
old-english	27.8	41.4	1.3	42.4
polish	73.5	56.9	1.151	55.1
russian	76.4	60.7	1.212	59.9

Table 6: Results for all languages for high data.

languages	BA	Acc - dev	Avg. L.D. -dev	Acc -test
adyghe	91.6	99.6	0.008	99.8
albanian	79.5	97.5	0.044	96.5
arabic	47.1	84.5	0.538	2.9
armenian	86.6	94.4	0.133	93.7
asturian	95.2	98.3	0.037	98.5
azeri	70	99	0.01	99
bashkir	90.7	99.8	0.003	99.7
basque	7.3	98.4	0.033	98
belarusian	41	88.4	0.235	88.4
bengali	81	98	0.05	99
breton	73	89	0.24	93
bulgarian	89	93.9	0.115	94.5
catalan	95.7	97.6	0.062	98
classical-syriac	97	100	0	100
crimean-tatar	95	100	0	100
czech	90.6	88.4	0.26	88
danish	87	91.2	0.158	91.3
estonian	78	97.2	0.075	95.9
faroese	76.1	80	0.437	81.1
finnish	78	76	0.603	76.4
friulian	96	100	0	99
galician	95.1	98.7	0.02	98.7
georgian	93.9	97.7	0.058	97.3
greek	78.3	82.6	0.407	80.8
haida	66	100	0	100
hebrew	53.7	97	0.063	97.3
hindi	93	99.6	0.006	99.4
hungarian	68.8	82.7	0.392	82.3
icelandic	76.9	86.5	0.297	83.9
irish	53	68.9	1.062	67.2
italian	77.5	95.5	0.106	95.7
kabardian	86	100	0	99

khaling	53.7	99.3	0.016	98.4
kurmanji	92.9	92.1	0.127	93.5
ladin	92	98	0.07	98
latin	47.6	59.6	0.652	61.5
latvian	92.8	92.8	0.162	92.8
lithuanian	64.2	86.7	0.234	88.1
livonian	67	92	0.2	97
lower-sorbian	88.1	95.9	0.078	94.3
macedonian	91.2	96.1	0.111	94.6
maltese	16	80	0.37	69
middle-french	95.1	98.6	0.038	96.7
navajo	0	79.9	0.603	75.3
neapolitan	95	98	0.02	97
northern-sami	62.3	94	0.145	92.6
norwegian-bokmaal	91	90.1	0.159	89
norwegian-nynorsk	74.8	83.6	0.295	82.6
occitan	96	100	0	99
old-armenian	79.2	84.1	0.381	87.3
old-church-slavonic	80	94	0.11	97
old-french	80.7	83.8	0.431	86.5
old-saxon	60.1	97.1	0.055	96.3
pashto	72	100	0	100
persian	80.7	98.5	0.028	98.9
portuguese	96.7	97.7	0.06	98.5
quechua	95.1	99.4	0.009	99.4
romanian	79.8	80.8	0.742	79
sanskrit	80.6	92.4	0.151	92.1
serbo-croatian	83	83.4	0.439	84.1
slovak	83.1	91.3	0.164	92.8
slovene	84.9	94.5	0.113	34.6
sorani	63.6	88.4	0.206	89
spanish	92.4	93.2	0.166	93.4
swahili	0	87	0.36	100
swedish	84.7	85.2	0.325	87.2
tatar	95	100	0	99
turkish	73.2	97.4	0.088	97.3
ukrainian	86.3	91.1	0.151	92.5
urdu	95.9	99.7	0.004	99.7
uzbek	96	99.3	0.012	98
venetian	93	95	0.14	99.1
votic	34	68	0.58	66
welsh	72	96	0.08	94
west-frisian	67	80	0.43	74
yiddish	94	98	0.08	99
zulu	0.2	97.8	0.055	97.2
dutch	87.7	94.8	0.102	93.1
english	95.9	94.7	0.133	95.8
french	83	80	0.514	81.1
german	81.1	85.8	0.395	83.7
kannada	66	99	0.01	100
north-frisian	37	93	0.15	96

old-english	40.9	84.9	0.307	83.4
polish	87.1	85.5	0.379	82.8
russian	86.5	84.4	0.509	85.4