Seq2Seq or Perceptrons for robust Lemmatization. An empirical examination.

Tobias Pütz, Daniël de Kok, Sebastian Pütz, Erhard Hinrichs

SFB833 A3
University of Tübingen

TLT17 Oslo, December 14, 2018

What's the takeaway?

- **High performance Lemmatization:** both linear edit-tree classification and neural Seq2Seq methods are highly competitive methods for lemmatization.
- **Classification:** predefined search space + explicit vocabulary help with language variation
- **Seq2Seq:** fine-grained character representations allow for better generalization to unknown items

Our Background

TüBa-D/DP: automatically annotated treebank for German:

- **28.6 billion** tokens (Wikipedia, TAZ, Europarl, Common Crawl)
- **Annotations:** dependency relations, topological fields, POS, morphological tags and **lemmas**
- **Current lemmatizer:** Lemming (Müller et al., 2015)
- **Task at hand:** examine and compare robustness of recent neural methods with Lemming

Data

1. **Form:** *interessante*
   Features: Adjective.accusative.plural.feminine
   Lemma: *interessant* 'interesting'

2. **Form:** *führten*
   Features: Finite Verb.3.indicative.past
   Lemma: *führen* 'to lead'

3. **Form:** *gelacht*
   Features: Perfect Participle
   Lemma: *lachen* 'to laugh'

Data

1. **Form:** *interessante*
   Features: Adjective.accusative.plural.feminine
   Lemma: *interessant* ‘interesting’

2. **Form:** *führten*
   Features: Finite Verb.3.indicative.past
   Lemma: *führen* ‘to lead’

3. **Form:** *gelacht*
   Features: Perfect Participle
   Lemma: *lachen* ‘to laugh’

- **Irregular forms** cannot be predicted and need to be dealt with seperately.

Dealing with non-standard language
Data

In compliance with TüBa-D/Z guidelines:

- **Spelling errors** in the form should be corrected in the lemma:
  - **uneingeschänkt* → *uneingeschränkt* ‘unlimited’

Dealing with non-standard language

Data

In compliance with TüBa-D/Z guidelines:

- **Spelling errors** in the form should be corrected in the lemma:
    - **uneingeschänkt* → *uneingeschränkt* 'unlimited'
- **Language variation** should be reduced to the lemma of the canonical form with a trailing underscore:
    - *koscht* → *kosten_* 'to cost'

Edit-scripts and Seq2Seq

Edit-script classifier

Chrupała (2006); Chrupała et al. (2008)

Müller et al. (2015)

ge	arbeite	*t*
	arbeite	*n*
del(**ge**)	*match*	*subst*(**t**,**n**)

1. **For each form-lemma pair:**
   - derive edit-scripts by aligning form-lemma pairs

Edit-script classifier

Chrupała (2006); Chrupała et al. (2008)

Müller et al. (2015)

ge	arbeite	*t*
	arbeite	*n*
del(**ge**)	*match*	*subst*(**t**,**n**)

1. **For each form-lemma pair:**
   - derive edit-scripts by aligning form-lemma pairs
2. **For each form:**
   1. create candidate set by applying all edit-scripts
   2. perform classification over candidate set

Edit-script classifier

Chrupała (2006); Chrupała et al. (2008)

Müller et al. (2015)

ge	arbeite	*t*
	arbeite	*n*
del(**ge**)	*match*	*subst*(**t**,**n**)

1. **For each form-lemma pair:**
   - derive edit-scripts by aligning form-lemma pairs
2. **For each form:**
   1. create candidate set by applying all edit-scripts
   2. perform classification over candidate set

- include **candidate lemma features**
- mostly **linear classifiers**
- rely on **engineered features**

Seq2Seq

Sutskever et al. (2014)

Der Hund jagte den Hasen.

The dog chased the rabbit.

Seq2Seq: state-of-the-art results on many sequence transduction tasks.

Seq2Seq

Sutskever et al. (2014)

Der Hund jagte den Hasen.

The dog chased the rabbit.

Seq2Seq: state-of-the-art results on many sequence transduction tasks.

- **little/no feature engineering:** features other than surface form are mostly basic linguistic units

Seq2Seq

Sutskever et al. (2014)

Der Hund jagte den Hasen.

The dog chased the rabbit.

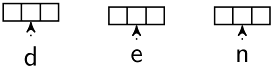
Seq2Seq: state-of-the-art results on many sequence transduction tasks.

- **little/no feature engineering:** features other than surface form are mostly basic linguistic units
- **fine-grained:** character-based representation helps to generalize to unseen combinations

Seq2Seq in 5 minutes

The encoder

Seq2Seq in 5 minutes

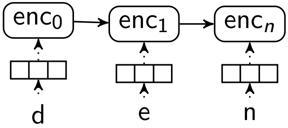


Encodes a form of arbitrary length into a fixed size vector:

- **Input:** characters mapped to real valued vectors (**embeddings**)

The encoder

Seq2Seq in 5 minutes

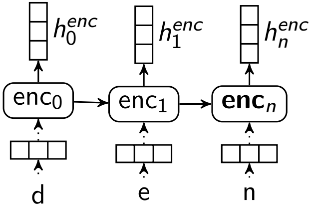


Encodes a form of arbitrary length into a fixed size vector:

- **Input:** characters mapped to real valued vectors (**embeddings**)
- **Processor:** Recurrent Neural Network
  - reads **one character per step**
  - **maintains hidden state** by composing it from current input and previous state

The encoder

Seq2Seq in 5 minutes



Encodes a form of arbitrary length into a fixed size vector:

- **Input:** characters mapped to real valued vectors (**embeddings**)
- **Processor:** Recurrent Neural Network
  - reads **one character per step**
  - **maintains hidden state** by composing it from current input and previous state
- **Output:**
  - intermediate states
  - final state

The decoder

Seq2Seq in 5 minutes

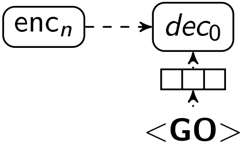
$\boxed{\mathbf{enc}_n} \dashrightarrow \boxed{dec_0}$

Decodes the final state of the encoder into a lemma of arbitrary length:

- **Initial state:** final state of the encoder

The decoder

Seq2Seq in 5 minutes

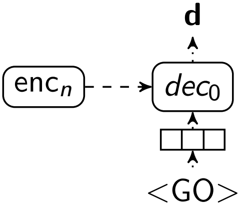


Decodes the final state of the encoder into a lemma of arbitrary length:

- **Initial state:** final state of the encoder
- **First input:** a special start symbol

The decoder

Seq2Seq in 5 minutes

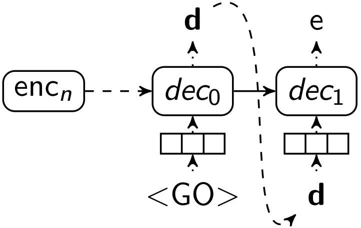


Decodes the final state of the encoder into a lemma of arbitrary length:

- **Initial state:** final state of the encoder
- **First input:** a special start symbol
- **Output:** probability distribution over characters

The decoder

Seq2Seq in 5 minutes

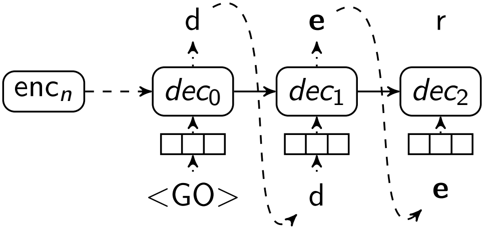


Decodes the final state of the encoder into a lemma of arbitrary length:

- **Initial state:** final state of the encoder
- **First input:** a special start symbol
- **Output:** probability distribution over characters
- **Subsequent inputs:** highest scoring character form previous step

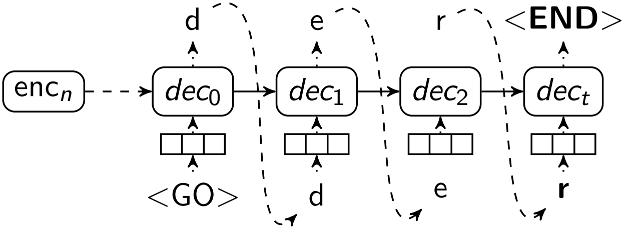
The decoder

Seq2Seq in 5 minutes



Decodes the final state of the encoder into a lemma of arbitrary length:

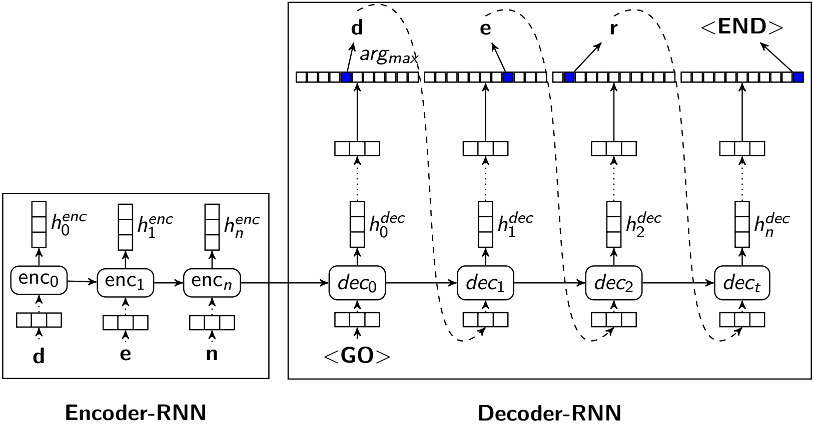
- **Initial state:** final state of the encoder
- **First input:** a special start symbol
- **Output:** probability distribution over characters
- **Subsequent inputs:** highest scoring character form previous step



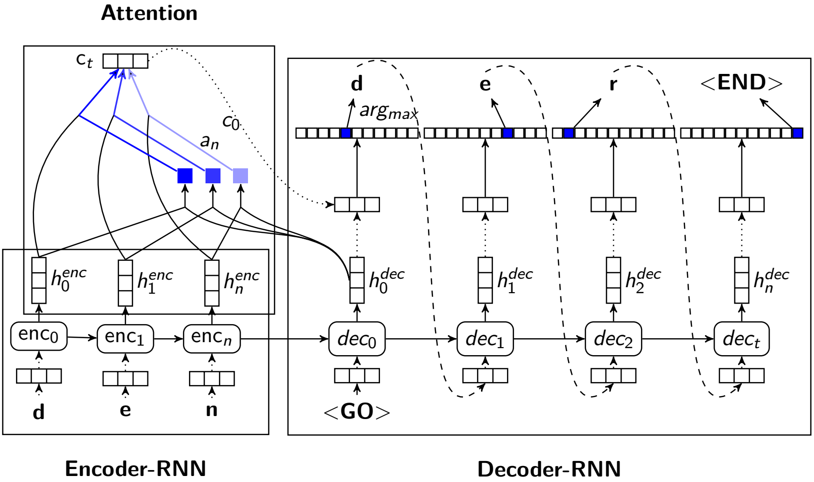
Decodes the final state of the encoder into a lemma of arbitrary length:

- **Initial state:** final state of the encoder
- **First input:** a special start symbol
- **Output:** probability distribution over characters
- **Subsequent inputs:** highest scoring character form previous step
- **Terminates:** when the end symbol is predicted

Seq2Seq in 5 minutes



Seq2Seq in 5 minutes



Bahdanau et al. (2014); Luong et al. (2015)

Setup

We compare three models on German:

- **Ohnomore**$_{seq2seq}$ (Oh-Morph): attentional Seq2Seq over characters with a morphologically informed decoder
- **Lemming**: linear edit-tree classifier (Müller et al., 2015)
  - **Lemming-Base**: built-in features
  - **Lemming-List**: built-in features + external word list
- Further information on the setup can be found in the TLT paper and my BA thesis (Pütz, 2018).[1]

[1] available at `http://sfs.uni-tuebingen.de/~tpuetz/`

Types

Setup

- **Type:** a unique combination of form, features and lemma
- **Example:**

Form:	*interessante*
Features:	Adjective.accusative.plural.feminine
Lemma:	*interessant* 'interesting'

- **Type:** a unique combination of form, features and lemma
- **Example:**

Form:	*interessante*
Features:	Adjective.accusative.plural.feminine
Lemma:	*interessant* ‘interesting’

- Train and test set are **disjoint sets of types** to ensure that the models are not just memorizing form-feature-lemma combinations.

Results

General Results - TüBa-D/Z

- Accuracy on types:

Model	**TüBa-D/Z**
Oh-Morph	97.00%
Lemming-Base	96.78%
Lemming-List	**97.02%**

- slight difference between **Lemming-List** and **Oh-Morph**
- the extended vocabulary of **Lemming-List** provides a boost of 0.24% over **Lemming-Base**

Out-of-vocabulary
Analysis

- **Vocabs:** *train* and *list*

Out-of-vocabulary

Analysis

- **Vocabs:** *train* and *list*
- **Oh-Morph:** highest accuracy across all out-of-vocabulary items

Out-of-vocabulary
Analysis

- **Vocabs:** *train* and *list*
- **Oh-Morph:** highest accuracy across all out-of-vocabulary items
- **Lemming:** dependence on completeness of vocabulary
  - **List:** worse performance than **Lemming-Base** on out-of-list items
  - **Base:** similar to **Oh-Morph** on out-of-list items but falls behind on out-of-train-vocab items

Partitions

Analysis

We analyze the two partitions of the test results:

1. **Shared:** all three models produced the same lemma

Examples:

	Form	Lemma	Oh-Morph	Lemming-List	Lemming-Base
Shared	*gearbeitet*	*arbeiten* ‘to work’	**arbeiten**	**arbeiten**	**arbeiten**

Partitions

Analysis

We analyze the two partitions of the test results:

1. **Shared:** all three models produced the same lemma
2. **Unique:** at least one model made a unique prediction

Examples:

	Form	Lemma	Oh-Morph	Lemming-List	Lemming-Base
Shared	*gearbeitet*	*arbeiten* ‘to work’	**arbeiten**	**arbeiten**	**arbeiten**
Unique	*Soloalben*	*Soloalbum* ‘solo album’	Soloalbum	Soloalbum	***Soloalbe**

Word list: helps with misspelled forms but misspelling is still the biggest error source.

- **unique: Lemming-List** 50% less errors than **Lemming-Base** and **Oh-Morph**

- **shared:** misspelling $>$ 30% of errors

Language Variation

Analysis

Explicit vocabulary helps but no model suited for the task:

- **unique:**
  - **both Lemming** models 70% error rate
  - **Oh-Morph** 85% error rate
- **shared:** 68% error rate

Language Variation

Analysis

Explicit vocabulary helps but no model suited for the task:

- **unique:**
  - **both Lemming** models 70% error rate
  - **Oh-Morph** 85% error rate
- **shared:** 68% error rate
- more domain specific training data and sentential context necessary

Language Variation
Analysis

Explicit vocabulary helps but no model suited for the task:

- **unique:**
  - **both Lemming** models 70% error rate
  - **Oh-Morph** 85% error rate
- **shared:** 68% error rate
- more domain specific training data and sentential context necessary
- fuzzy line between spelling errors and language variation

Conclusion and Outlook

Conclusion

- Seq2Seq and edit-tree classifier have different strengths
- **featurizing** a candidate set helps with spelling variation
- **character-based** Seq2Seq generalizes well to unseen items

Outlook

- use the complementary strengths in an **ensemble**
- joint lemmatization and text normalization

Work in progress

Two directions:

- **combination of bi- and uni-directionality** in the encoder gives promising results
- **neural edit-tree classifier**

Thank you!

Partitions

Analysis

1. **Shared:** (207,627 types) all models produced the same lemma:
   - **Error rate:** 1.60% (# 3322)
2. **Unique:** (6,078 types) at least one model produced a unique lemma
   - **Oh-Morph:** 50.80% (# 3087)
   - **Lemming-Base:** 58.65% (# 3565)
   - **Lemming-List:** 50.20% (# 3051)

Unknowns

Vocab	Type	Oh-Morph	Lemming-Base	Lemming-List
Train	Form	**95.74%**	95.21%	95.62%
	Lemma	**96.32%**	96.04%	95.98%
List	Form	**94.34%**	94.27%	94.20%
	Lemma	**96.48%**	96.47%	95.70%

Irregular Forms

Examples:

1. **Form:** *bot*
   Features: Finite Verb.3.indicative.past
   Lemma: *bieten* 'to bid'

2. **Form:** *darf*
   Features: Finite Verb.3.indicative.past
   Lemma: *dürfen* 'be allowed to'

What people do:

1. **Dictionary:** complement lemmatizer with a dictionary
2. **Overlapping train-validation sets:** effectively treating the training set as a dictionary

- **Both:** coverage is limited to dictionary / training data