Deep-speare: A Joint Neural Model of Poetic Language, Meter and Rhyme

Jey Han Lau\textsuperscript{1,2}, Trevor Cohn\textsuperscript{2}, Timothy Baldwin\textsuperscript{2}, Julian Brooke\textsuperscript{3}, and Adam Hammond\textsuperscript{4}

\textsuperscript{1} IBM Research Australia
\textsuperscript{2} School of CIS, The University of Melbourne
\textsuperscript{3} University of British Columbia
\textsuperscript{4} Dept of English, University of Toronto

July 17, 2018
Creativity

- Can machine learning models be creative?

- Can these models compose novel and interesting narrative?

- Creativity is a hallmark of intelligence — it often involves blending ideas from different domains.

- We focus on sonnet generation in this work.
Shall I compare thee to a summer’s day?
Thou art more lovely and more temperate:
Rough winds do shake the darling buds of May,
And summer’s lease hath all too short a date:

A distinguishing feature of poetry is its aesthetic forms, e.g. rhyme and rhythm/meter.

Rhyme: \{day, May\}; \{temperate, date\}.

Stress (pentameter):

\[ S^- S^+ S^- S^+ S^- S^+ S^- S^+ S^- S^+ \]

Shall I compare thee to a summer’s day?
Modelling Approach

- We treat the task of poem generation as a constrained language modelling task.

- Given a rhyming scheme, each line follows a canonical meter and has a fixed number of stresses.

- We focus specifically on sonnets as it is a popular type of poetry (sufficient data) and has regular rhyming (ABAB, AABB or ABBA) and stress pattern (iambic pentameter).

- We train an unsupervised model of language, rhyme and meter on a corpus of sonnets.
We first create a generic poetry document collection using GutenTag tool, based on its inbuilt poetry classifier.

We then extract word and character statistics from Shakespeare’s 154 sonnets.

We use the statistics to filter out all non-sonnet poems, yielding our sonnet corpus.

<table>
<thead>
<tr>
<th>Partition</th>
<th>#Sonnets</th>
<th>#Words</th>
</tr>
</thead>
<tbody>
<tr>
<td>Train</td>
<td>2685</td>
<td>367K</td>
</tr>
<tr>
<td>Dev</td>
<td>335</td>
<td>46K</td>
</tr>
<tr>
<td>Test</td>
<td>335</td>
<td>46K</td>
</tr>
</tbody>
</table>
Model Architecture

(a) Language model
(b) Pentameter model
(c) Rhyme model
Language Model (LM)

- LM is a variant of an LSTM encoder–decoder model with attention.
- Encoder encodes preceding contexts, i.e. all sonnet lines before the current line.
- Decoder decodes one word at a time for the current line, while attending to the preceding context.
- Preceding context is filtered by a selective mechanism.
- Character encodings are incorporated for decoder input words.
- Input and output word embeddings are tied.
Pentameter Model (PM)

- PM is designed to capture the alternating stress pattern.

- Given a sonnet line, PM learns to attend to the appropriate characters to predict the 10 binary stress symbols sequentially.

<table>
<thead>
<tr>
<th>T</th>
<th>Attention</th>
<th>Prediction</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td><em>Shall I compare thee to a summer’s day?</em></td>
<td>$S^-$</td>
</tr>
<tr>
<td>1</td>
<td><em>Shall I compare thee to a summer’s day?</em></td>
<td>$S^+$</td>
</tr>
<tr>
<td>2</td>
<td><em>Shall I compare thee to a summer’s day?</em></td>
<td>$S^-$</td>
</tr>
<tr>
<td>3</td>
<td><em>Shall I compare thee to a summer’s day?</em></td>
<td>$S^+$</td>
</tr>
<tr>
<td></td>
<td>...</td>
<td></td>
</tr>
<tr>
<td>8</td>
<td><em>Shall I compare thee to a summer’s day?</em></td>
<td>$S^-$</td>
</tr>
<tr>
<td>9</td>
<td><em>Shall I compare thee to a summer’s day?</em></td>
<td>$S^+$</td>
</tr>
</tbody>
</table>
Pentameter Model (PM)

- PM fashioned as an encoder–decoder model.
- Encoder encodes the characters of a sonnet line.
- Decoder attends to the character encodings to predict the stresses.
- Decoder states are not used in prediction.
- Attention networks focus on characters whose position is monotonically increasing.
- In addition to cross-entropy loss, PM is regularised further with two auxilliary objectives that penalise repetition and low coverage.
Pentameter Model (PM)
Rhyme Model

- We learn rhyme in an unsupervised fashion for 2 reasons:
  - Extendable to other languages that don't have pronunciation dictionaries;
  - The language of our sonnets is not Modern English, so contemporary pronunciation dictionaries may not be accurate.

- Assumption: rhyme exists in a quatrain.

- Feed sentence-ending word pairs as input to the rhyme model and train it to separate rhyming word pairs from non-rhyming ones.
Rhyme Model

Shall I compare thee to a summer’s day? Shall I compare thee to a summer’s day?
Thou art more lovely and more temperate:
Rough winds do shake the darling buds of May,
And summer’s lease hath all too short a date:

\[ Q = \{ \cos(\bar{u}_t, \bar{u}_r), \cos(\bar{u}_t, \bar{u}_{r+1}), \cos(\bar{u}_t, \bar{u}_{r+2}) \} \]
\[ \mathcal{L}_{rm} = \max(0, \delta - \text{top}(Q, 1) + \text{top}(Q, 2)) \]

- top\( (Q, k) \) returns the \( k \)-th largest element in \( Q \).

- Intuitively the model is trained to learn a sufficient margin that separates the best pair from all others, with the second-best being used to quantify all others.
Joint Training

- All components trained together by treating each component as a sub-task in a multi-task learning setting.

- Although the components (LM, PM and RM) appear to be disjointed, shared parameters allow the components to mutually influence each other during training.

- If each component is trained separately, PM performs poorly.
Model Architecture

(a) Language model

(b) Pentameter model

(c) Rhyme model
Evaluation: Crowdworkers

Crowdworkers are presented with a pair of poems (one machine-generated and one human-written), and asked to guess which is the human-written one.

- **LM**: vanilla LSTM language model;
- **LM**: LSTM language model that incorporates both character encodings and preceding context;
- **LM**+PM+RM: the full model, with joint training of the language, pentameter and rhyme models.
Evaluation: Crowdworkers (2)

<table>
<thead>
<tr>
<th>Model</th>
<th>Accuracy</th>
</tr>
</thead>
<tbody>
<tr>
<td>LM</td>
<td>0.742</td>
</tr>
<tr>
<td>LM**</td>
<td>0.672</td>
</tr>
<tr>
<td>LM**+PM+RM</td>
<td>0.532</td>
</tr>
<tr>
<td>LM**+RM</td>
<td>0.532</td>
</tr>
</tbody>
</table>

- Accuracy improves $\text{LM} < \text{LM}^{**} < \text{LM}^{**} + \text{PM} + \text{RM}$, indicating generated quatrains are less distinguishable.

- Are workers judging poems using just rhyme?

- Test with $\text{LM}^{**} + \text{RM}$ reveals that’s the case.

- Meter/stress is largely ignored by laypersons in poetry evaluation.
Evaluation: Expert

<table>
<thead>
<tr>
<th>Model</th>
<th>Meter</th>
<th>Rhyme</th>
<th>Read.</th>
<th>Emotion</th>
</tr>
</thead>
<tbody>
<tr>
<td>LM</td>
<td>4.00±0.73</td>
<td>1.57±0.67</td>
<td>2.77±0.67</td>
<td>2.73±0.51</td>
</tr>
<tr>
<td>LM**</td>
<td>4.07±1.03</td>
<td>1.53±0.88</td>
<td>3.10±1.04</td>
<td>2.93±0.93</td>
</tr>
<tr>
<td>LM**+PM+RM</td>
<td>4.10±0.91</td>
<td>4.43±0.56</td>
<td>2.70±0.69</td>
<td>2.90±0.79</td>
</tr>
<tr>
<td>Human</td>
<td>3.87±1.12</td>
<td>4.10±1.35</td>
<td>4.80±0.48</td>
<td>4.37±0.71</td>
</tr>
</tbody>
</table>

- A literature expert is asked to judge poems on the quality of meter, rhyme, readability and emotion.

- Full model has the highest meter and rhyme ratings, even higher than human, reflecting that poets regularly break rules.

- Despite excellent form, machine-generated poems are easily distinguished due to lower emotional impact and readability.

- Vanilla language model (LM) captures meter surprisingly well.
We introduce a joint neural model that learns language, rhyme and stress in an unsupervised fashion.

We encode assumptions we have about the rhyme and stress in the architecture of the network.

Model can be adapted to poetry in other languages.

We assess the quality of generated poems using judgements from crowdworkers and a literature expert.

Our results suggest future research should look beyond forms, towards the substance of good poetry.

Code and data: https://github.com/jhlau/deepspeare
in darkness to behold him, with a light
and him was filled with terror on my breast
and saw its brazen ruler of the night
but, lo! it was a monarch of the rest
Questions?