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

Jey Han Lau[1,2], Trevor Cohn[2], Timothy Baldwin[2],
Julian Brooke[3], and Adam Hammond[4]

[1] IBM Research Australia
[2] School of CIS, The University of Melbourne
[3] University of British Columbia
[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.

Sonnets

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):

$$\begin{array}{cccccccccc} S^- & S^+ & S^- & S^+ & S^- & S^+ & S^- & S^+ & S^- & S^+ \\ \textit{Shall} & \textit{I} & \multicolumn{2}{c}{\textit{compare}} & \textit{thee} & \textit{to} & \textit{a} & \multicolumn{2}{c}{\textit{summer's}} & \textit{day?} \end{array}$$

Modelling Approach

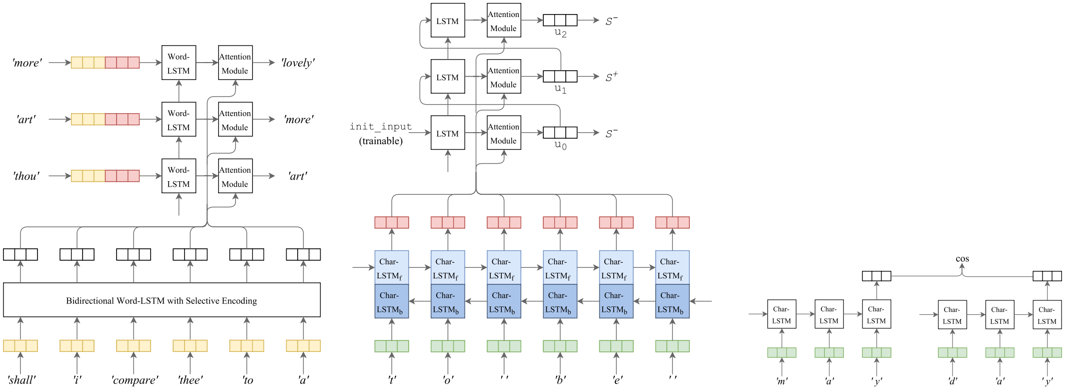
- We treat the task of poem generation as a constrained language modelling task.
- Given a rhyming scheme, each line follows a canonical meter and has a fixed number of stresses.
- We focus specifically on sonnets as it is a popular type of poetry (sufficient data) and has regular rhyming (ABAB, AABB or ABBA) and stress pattern (iambic pentameter).
- We train an unsupervised model of language, rhyme and meter on a corpus of sonnets.

Sonnet Corpus

- 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.

Partition	**#Sonnets**	**#Words**
Train	2685	367K
Dev	335	46K
Test	335	46K

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)

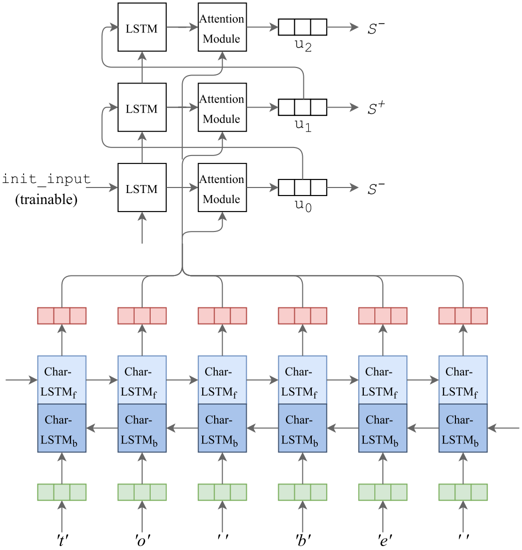
- PM is designed to capture the alternating stress pattern.
- Given a sonnet line, PM learns to attend to the appropriate characters to predict the 10 binary stress symbols sequentially.

T	Attention	Prediction
0	*Shall I compare thee to a summer's day?*	S^-
1	*Shall I compare thee to a summer's day?*	S^+
2	*Shall I compare thee to a summer's day?*	S^-
3	*Shall I compare thee to a summer's day?*	S^+
	...	
8	*Shall I compare thee to a summer's day?*	S^-
9	*Shall I compare thee to a summer's day?*	S^+

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? $\overline{\mathbf{u}}_t$
Thou art more lovely and more temperate: $\overline{\mathbf{u}}_r$
Rough winds do shake the darling buds of May, $\overline{\mathbf{u}}_{r+1}$
And summer's lease hath all too short a date: $\overline{\mathbf{u}}_{r+2}$

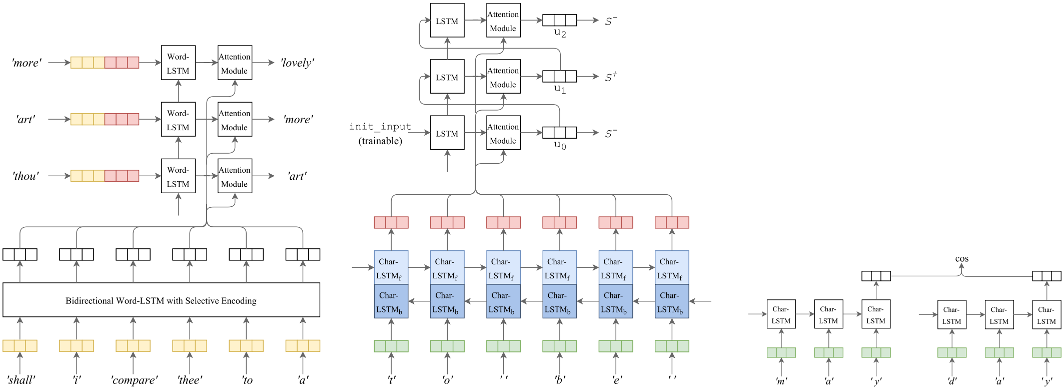
$$
\begin{aligned}
Q &= \{\cos(\overline{\mathbf{u}}_t, \overline{\mathbf{u}}_r), \cos(\overline{\mathbf{u}}_t, \overline{\mathbf{u}}_{r+1}), \cos(\overline{\mathbf{u}}_t, \overline{\mathbf{u}}_{r+2})\} \\
\mathcal{L}_{rm} &= \max(0, \delta - \mathrm{top}(Q, 1) + \mathrm{top}(Q, 2))
\end{aligned}
$$

- $\mathrm{top}(Q, k)$ 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)

Model	Accuracy
LM	0.742
LM**	0.672
LM**+PM+RM	0.532
LM**+RM	0.532

- Accuracy improves LM $<$ LM** $<$ LM**+PM+RM, indicating generated quatrains are less distinguishable.
- Are workers judging poems using just rhyme?
- Test with LM**+RM reveals that's the case.
- Meter/stress is largely ignored by laypersons in poetry evaluation.

Evaluation: Expert

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

- 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.

Summary

- 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`

"Untitled"

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?