Generating Personalized Chat Replies

Kristen Aw   jiayuaw@stanford.edu

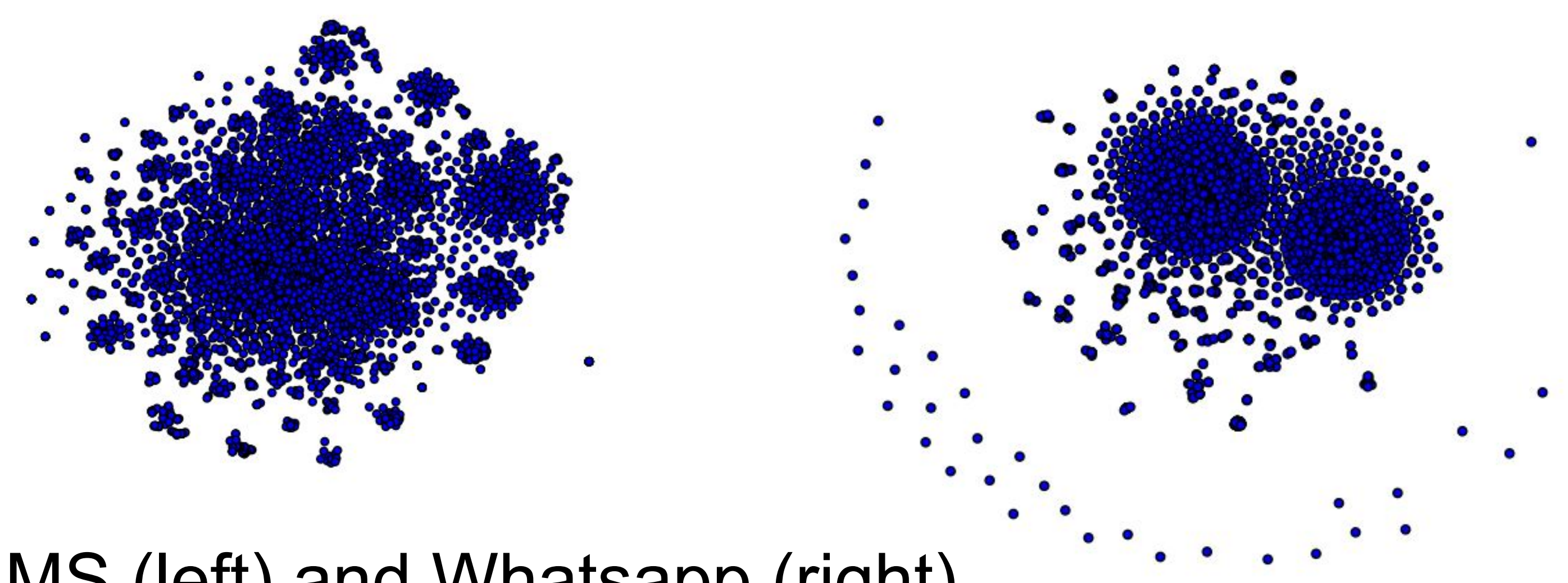
Overview

Problem: Automating replies to chat messages in the user's voice.
Solution: Classify the incoming message, pick a generator trained for that class, create the response.
Results: Generated replies that were plausible ~30% of the time.

Data

Personal: SMS (4.8k) and Whatsapp messages (3.6k).
Public: Marsan Ma Twitter chats (700k+)
Train/dev/test split: 0.8/0.1/0.1

Features



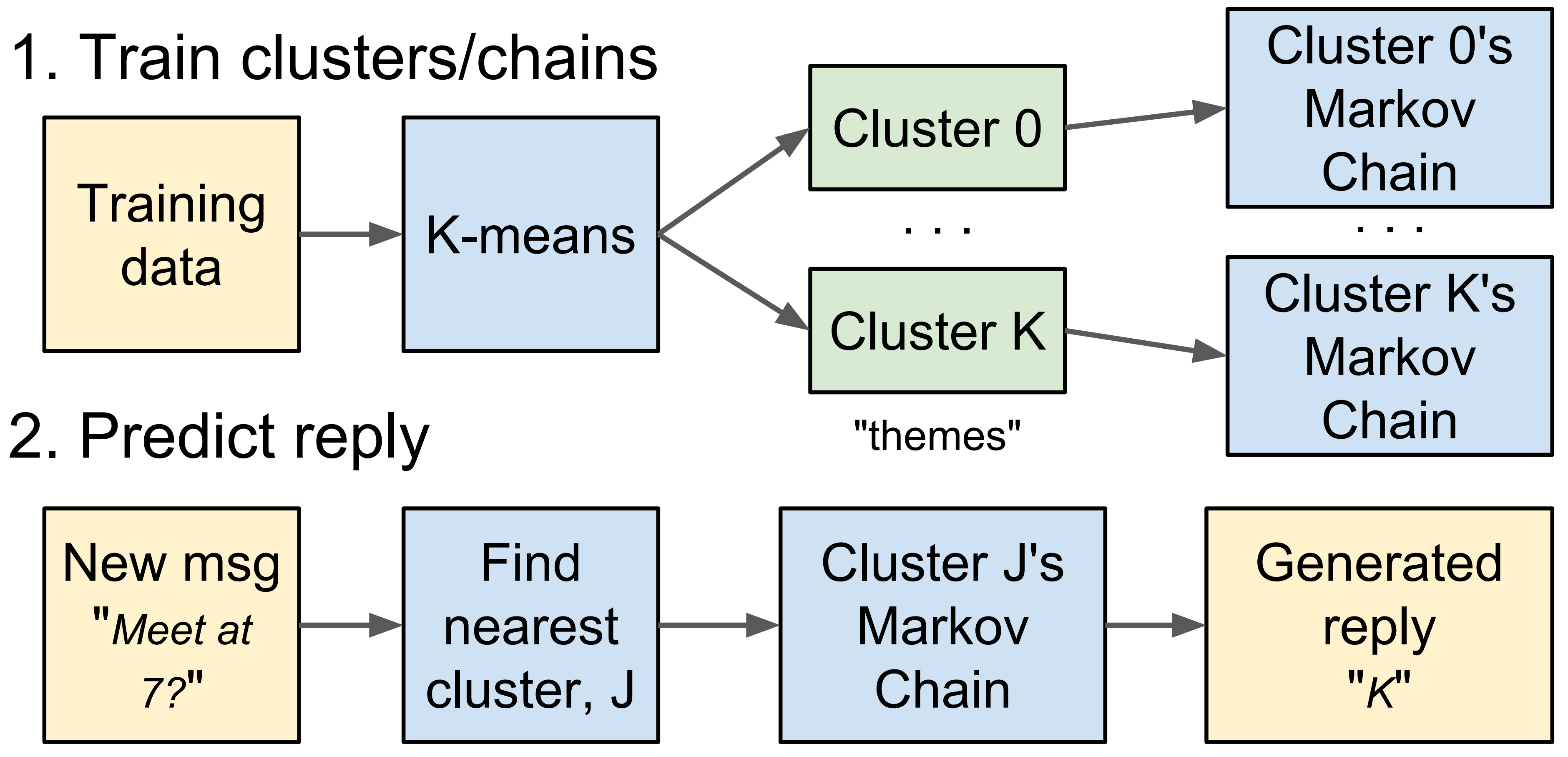
SMS (left) and Whatsapp (right) messages visualized with TSNE.

Raw input	Feature	Purpose
Each message	BOW vector	Classify messages based on their words
Each word	Word index	To transform into sequences / embeddings; count word occurrences

References: Sutskever et al- Sequence to Sequence Learning with Neueral Networks; Vinyals, Le- A Neural Conversational Model; Greff et al- LSTM: A Search Space Odyssey. Suriyadeepan- Practical Seq2Seq, Google- Tensorflow NMT tutorial.

Models

K-means + Markov Chain (K+MC)



1. Train clusters/chains

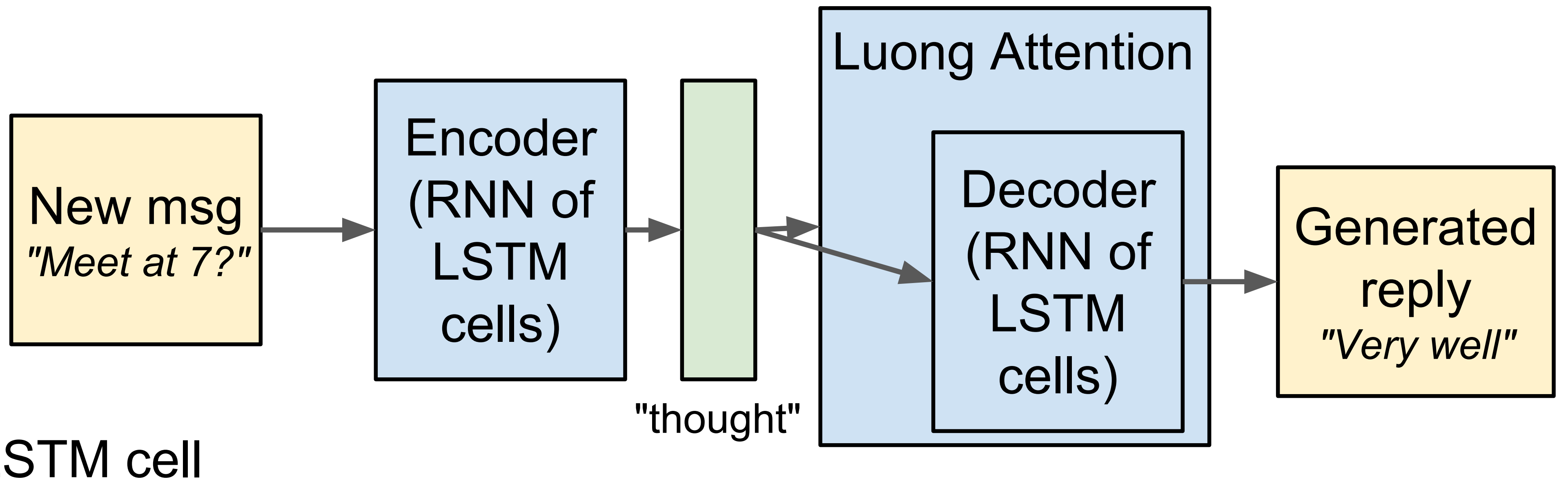
Training data → K-means → Cluster 0 … Cluster K ("themes") → Cluster 0's Markov Chain … Cluster K's Markov Chain

2. Predict reply

New msg *"Meet at 7?"* → Find nearest cluster, J → Cluster J's Markov Chain → Generated reply *"K"*

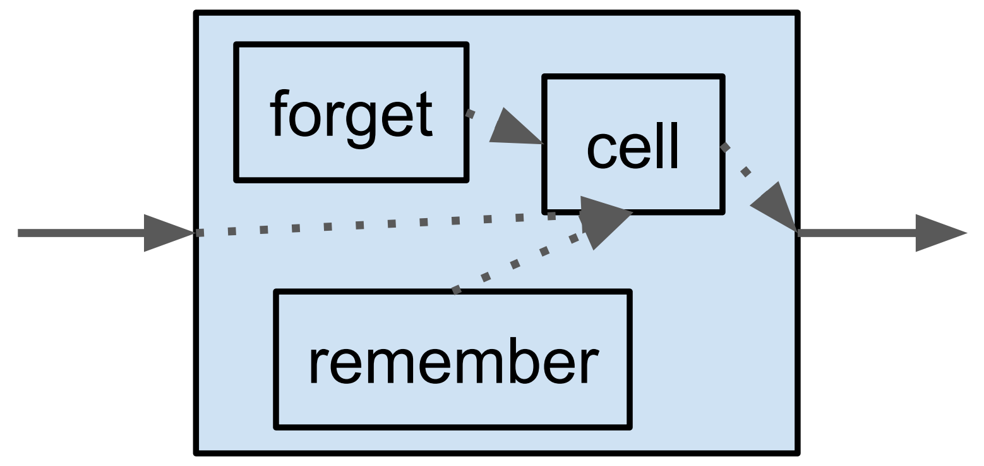
Alternative: Seq2Seq+Attention (s2s+A)

State-of-the-art, generates meaningful sentences



New msg *"Meet at 7?"* → Encoder (RNN of LSTM cells) → "thought" → Luong Attention: Decoder (RNN of LSTM cells) → Generated reply *"Very well"*

LSTM cell



BUT: Needs lots of data. On personal dataset, performed poorly regardless of: regularization; number of layers/cells; using GRU or LSTM cell

Alternative (Baseline): K-means (K)

Like K+MC, but pick any response from nearest cluster. Mostly irrelevant responses, but **showed that messages could be clustered meaningfully** (k=11 better than lower k's). **Clusters had themes** like "making plans" and "catching up".

Results

"**Plausiblity**": Whether the reply (1) sounds like user; (2) is relevant to the input message. Human evaluated.

Model	Example	Sounds like user	Input-relevant
K	x: ok cool. hopping in singles y: i'm on the right hand side h: ohh why is there chocolate? thanks!	1.00	0.17
K+MC	x: how's the price? y: yay april then! h: 15-20%? i'm really it's etc://	**0.33**	**0.34**
s2s+A (w/ personal data)	x: eat first or swim? y: yep h: 20 if if if korean korean cost	0.06	0.12
s2s+A (w/ public data)	x: busy at work? y: been playing com games ohgod h: i love you	0.05	0.45

K+MC replies were more plausible than s2s+A's but sentence structures were worse.

Discussion

Generated replies were usually bad since
Both goals conflict with each other:
"Sounds like user" ~ 1 / "Relevance to input"
Hard to train: too little data. Models did worst: (a) MC on clusters vs all data; (b) s2s on personal vs public data.
Hard to evaluate: loss/ BLEU/ perplexity does not account for relevance/ personality.

Future Work

Find data-cheap method to permute s2s+A generated replies to personalized voice. Tried RL (MDP) on word characters but this did poorly.