Natural Language Generation

Topic Coverage

- Survey in the State of the Art of Natural Language Generation by Gatt and Krahmer
  - Intro and NLG tasks ->Tianqi Wu
  - NLG Architecture and Approaches ->Jianing Zhou
  - Style Variation and Creative Text -> Max Fowler
  - Evaluation -> Ningkai Wu

- Multi-domain Neural Network Language Generation for Spoken Dialog Systems by Wen et al. -> Samuel Kriman

Intro and NLG

Presented By Tianqi Wu

What is NLG?

Generating text/speech from **all kinds** of data

What to say and **how** to express

- text-to-text generation
- data-to-text generation

Text-to-Text Generation

Input: existing (human-written) text

- Machine Translation
- Text Summarization
- Simplification of Complex Texts
- Grammar and Text Correction

Data-to-Text Generation *

Input: non-linguistic data

- Automated Journalism (earthquake)
- Soccer Reports
- Weather and Financial Reports

NLG Tasks - Subproblems

- Content Determination
- Text Structuring
- Sentence Aggregation
- Lexicalisation
- Referring Expression Generation
- Linguistic Realisation

Content Determination

Extract the information of **interest,** which involves **choices** of what information to keep and what to ignore.

Which information to generate given description of a sick baby:
It depends on your **communicative goal**

- The baby is being given morphine via an IV drop ← parents
- The baby's heart rate shows bradycardia's (low heart rate) ← doctors
- The baby's temperature is normal
- The baby is crying ← parents

Text Structuring -- Coherence

Ordering of sentences matters

Consider generating a weather report:

1. It will rain on Thursday
2. It will be sunny on Friday
3. Max temperature will be 10C on Thursday
4. Max temperature will be 15C on Friday

Which of the following order would you prefer?

(1234), (2341), (4321) Human readers prefer **(1234)**

Sentence Aggregation -- Conciseness

Grouping of sentences

Consider generating a weather report again:

1. It will rain on Saturday
2. It will be sunny on Sunday
3. Max temperature will be 10C on Saturday
4. Max temperature will be 15C on Sunday

How would you combine sentences?

(12)(34), (1)(23)(4) Human readers prefer **(12)(34)**

Sentence Aggregation -- Conciseness

Describing **fastest** hat-trick in the English Premier League:

(1)Sadio Mane scored for Southampton after 12 minutes and 22 seconds.
(2)Sadio Mane scored for Southampton after 13 minutes and 46 seconds.
(3)Sadio Mane scored for Southampton after 15 minutes and 18 seconds.

Aggregating to one sentence is more preferred:

(4)Sadio Mane scored three times for Southampton in less than three minutes.

Lexicalisation

Alternative Expressions Selection

Scoring in soccer report:

- to score a goal
- to have a goal noted
- to put the ball in the net

Domain-dependent

Consider describing heavy rain:

weather report:	see rainfall totals over three inches
voice assistant:	expect heavy rain
idiom:	It is raining dogs and cats

Referring Expression Generation

Creation of referring expressions that **identify specific entities**
Received most attention since it can be separated easily

- Pronouns:
  - Tom saw a movie. **It** is interesting.
- Definite noun:
  - Tom saw a movie. **The** movie is interesting.

...

Linguistic Realisation

Combination of selected words and phrases to form sentence

- Human-Crafted Templates
  - A $location $gender in $pronoun $age, has been diagnosed with coronavirus on $date
  - A Chicago woman in her 60s, has been diagnosed with coronavirus on Jan. 24

- Statistical Approaches *

Strategy & Tactics

“Strategy without tactics is the slowest route to victory.

Tactics without strategy is the noise before defeat.”

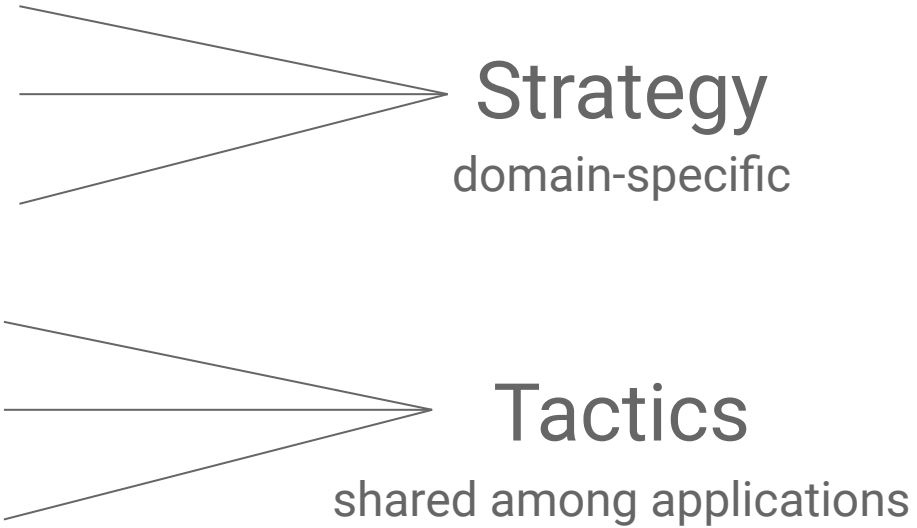
-- “The Art of War”

Strategy: long-term goal and how you are going to get there

Tactics: specific actions you are going to take along the way.

NLG Tasks

- Content Determination
- Text Structuring
- Sentence Aggregation
- Lexicalisation
- Referring Expression Gen
- Linguistic Realisation



Trend

Hand-crafted, rule-based, domain-dependent

Statistical, data-driven, domain-independent

(more efficient but output quality may be compromised)

NLG in Commercial Scenarios

Pure data-driven methods may not be favored.

- Inappropriate contents for certain readers
  - Siri used to help you find nearby bridges when you say “I want to jump off a bridge”

- Data not available in some domains

Recent Directions

Alternative approach: “end-to-end” machine learning
NLG is challenging: human languages are complex and ambiguous

Huge increase in available data and computing power
created new possibilities to:

- Image-to-text generations
- Applications to social media
- More industrial applications

NLG Architecture and Approaches

Presented By Jianing Zhou

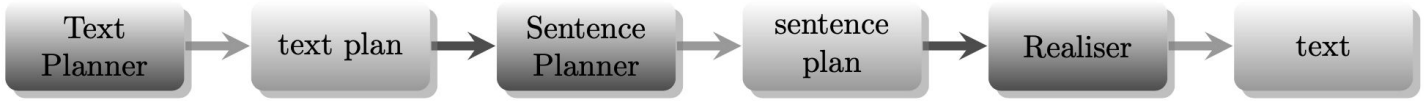
Outline

1. Modular Approaches
2. Planning-based Approaches
3. Other stochastic approaches to NLG

Modular Approaches

- Pipeline architecture
- Divide a task into several sub-tasks
- Different modules in the pipeline incorporate different subsets of the tasks
- Complete each task step by step and finally get the generated text

A classical modular architecture



1. Text Planner: combines content selection and text structuring;

   Mainly strategic generation, decides what to say

2. Sentence Planner: combines sentence aggregation, lexicalisation and referring expression generation;

   Decides how to say it

3. Realiser: perform linguistic realisation;

   generate the final sentences in a grammatically correct way.

Some other modular architectures

Mellish (2006):

‘object-and-arrows’ framework:

Different types of information flow between NLG sub-tasks can be accommodated.

Reiter (2007):

To accommodate systems in which input consists of raw (often numeric) data

Signal Analysis stage: detect basic patterns in the input data, Organize patterns into discrete events such as log files

Data Interpretation stage: map basic patterns and events into the messages and relationships that humans use

Another recent development

Proposed by Reiter (2007)

To accommodate systems in which input consists of raw (often numeric) data

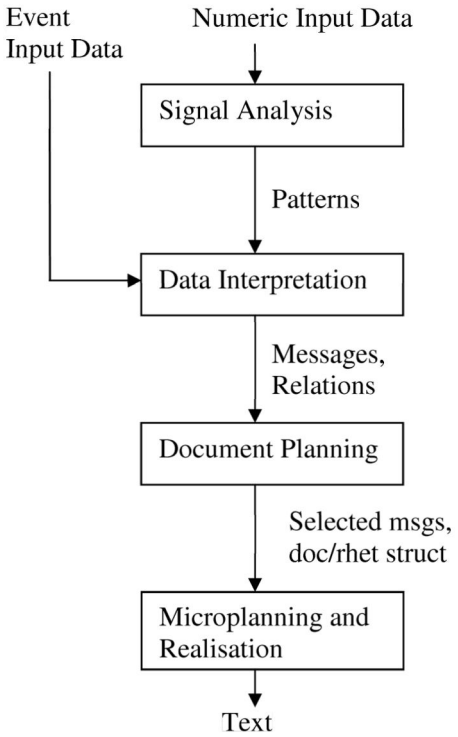
Main characteristic: input is unstructured and requires some preprocessing

Signal Analysis stage: detect basic patterns in the input data

Organize patterns into discrete events such as log files

Data Interpretation stage: map basic patterns and events into the messages

and relationships that humans use



Challenges

Two challenges associated with pipeline architectures

1. Generation gap: error propagation, early decisions in the pipeline have unforeseen consequences further

   downstream

2. Generating under constraints: e.g. the output cannot exceed a certain length. Possible at the realisation stage

   but harder at the earlier stages.

Alternative architectures motivated by these challenges:

1. Interactive design: feedback from a later module, Hovy, E. H. (1988).
2. Revision: feedback between modules under monitoring, Inui et al. (1992).

Planning-Based Approaches

- Planning Problem: identifying a sequence of one or more actions to satisfy a particular goal.
- Connection between planning and NLG:

  Text generation can be viewed as the execution of planned behaviour to achieve a communicative goal.

  State $\xrightarrow{\text{Action}}$ A new state

  A change in the context

  Current text $\xrightarrow{\text{Generation}}$ New text

- Methods:
  - Planning through the grammar
  - Planning using reinforcement learning

Planning through the grammar

Viewing linguistic structures as planning operators or actions

Consider the sentence *Mary likes the white rabbit.* We can represent the lexical item *likes* as follows:

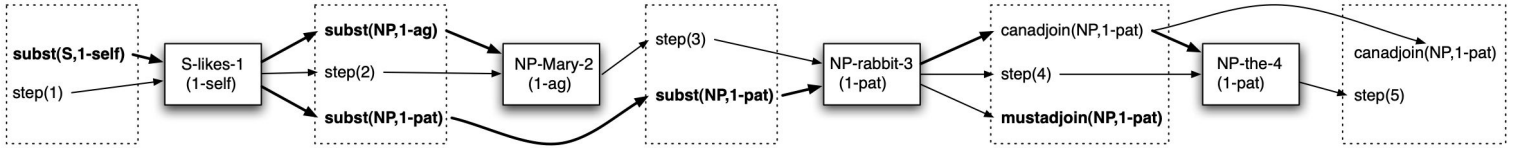
likes(u, x, y) action:
PRECONDITIONS:

- The proposition that x likes y is part of the knowledge base (i.e. the statement is supported);
- x is animate;
- The current utterance u can be substituted into the derivation S under construction;

EFFECTS:

- u is now part of S
- New NP nodes for x in agent position and y in patient position have been set up (and need to be filled).

Planning through the grammar



Having inserted likes as the sentence's main verb, we get two noun phrases which need to be filled by generating NPs for x and y.

Then, to generate noun phrases we get, we build referring expressions by associating further preconditions on the linguistic operators that will be incorporated in the referential NP.

Advantage: availability of a significant number of off-the-shelf planners.

> Once the nlg task is formulated in an appropriate plan description language, we can use any planner to generate text.

Planning through Reinforcement Learning

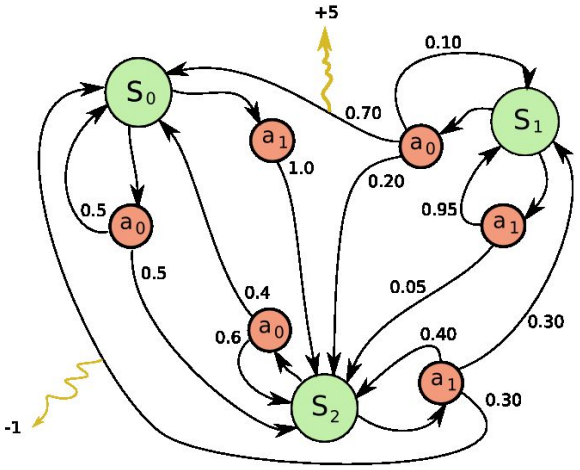
Main idea: planning a good solution to reach a communicative goal could be viewed as a stochastic optimisation problem.

So we can use RL to solve this problem. In this framework, generation can be modelled as a Markov Decision Process:

Each state is associated with possible actions;

Each state-action pair is associated with a probability of moving from a state at time t to a new state at t + 1 via action a;

Transitions are associated with rewards

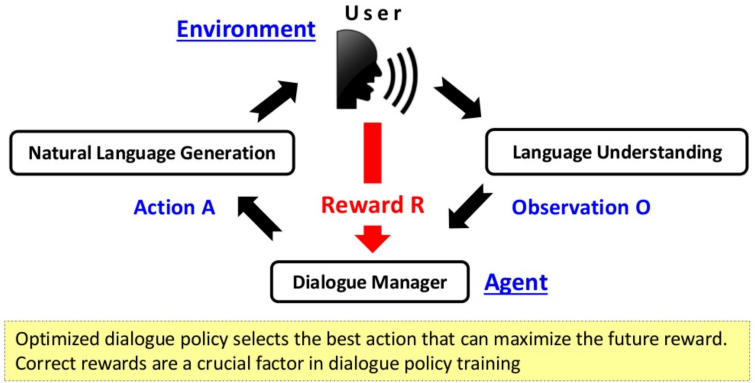


Plans corresponding to possible paths through the state space

Planning through Reinforcement Learning

Learning: simulations in which different generation strategies or policies are associated with different rewards

We want to find the best policy which maximizes rewards and use it to generate texts



Example: dialogue generation

Action: Generating sequences.

State: A state is denoted by the previous two dialogue turns.

Reward: Ease of answering, Information Flow and Semantic Coherence

Planning through Reinforcement Learning

Contribution:

1. Handling uncertainty in dynamic environments better by enabling adaptation in a changing context.

2. Exploring joint optimisation: the policy learned satisfies multiple constraints arising from different sub-tasks.

Other stochastic approaches to NLG

1. Acquiring Data
2. NLG as a Sequential, Stochastic Process
3. NLG as Classification and Optimisation
4. NLG as ‘Parsing’
5. Deep Learning Methods
6. Encoder-Decoder Architecture
7. Conditioned Language Models

Acquiring Data

Research on realisation often exploits the existence of treebanks from which input-output correspondences can be learned

The emergence of corpora of referring expressions has facilitated the development of probabilistic REG algorithms

Recent work on image-to-text generation has also benefited from the availability of large datasets.

Therefore, many tasks benefit from data sources and methods.

A promising trend is the introduction of statistical techniques that seek to automatically segment and align data and text

- Liang et al. (2009) proposed a model performing alignment by identifying regular co-occurrences of data and text
- Koncel-Kedziorski et al. (2014) go beyond this by proposing a model that exploits linguistic structure to align
- Mairesse and Young (2014) use crowd-sourcing techniques to elicit realisations for semantic/pragmatic inputs

More recent stochastic methods based on NN obviate the need for alignment

NLG as a Sequential, Stochastic Process

Given an alignment between data and text, one way of modelling the NLG process is to use sequential/pipeline arch

1. Using the statistical alignment to inform content selection
2. Use NLP techniques to acquire rules, templates or schemas to drive sentence planning and realisation.

- Oh and Rudnicky (2002) used Markov-based LM in content planning and realisation
- Ratnaparkhi (2000) used conditional LM to generate sentences by predicting the best word given both the preceding history and the semantic attributes that remain to be expressed
- Angeli et al. (2010) describe an end-to-end nlg system that maintains a separation between content selection, sentence planning and realisation, modelling each process as a sequence of decisions in a log-linear framework, where choices can be conditioned on arbitrarily long histories of previous decisions.

NLG as Classification and Optimisation

Classification: generation is ultimately about choice-making at multiple levels, so we use a cascade of classifiers, where the output is constructed incrementally, so that any classifier Ci uses as (part of) its input the output of a previous classifier Ci−i.

But the main problem is error propagation, Infelicitous choices will impact classification further downstream.

Solution: view generation as an optimisation problem, the best combination of decisions is sought in a space of possible combinations.

Optimisation:

1. Each nlg task is once again modelled as classification associated with a cost function.
2. Pairs of tasks which are strongly inter-dependent have a cost based on the joint probability of their labels
3. Seek the global labelling solution that minimizes the overall cost.

NLG as ‘Parsing’

Main idea: view generation as the inverse of semantic parsing

Example: WASP and WASPER-GEN

WASP: maximize the probability of a meaning representation given a sentence to learn a parser

WASPER-GEN: seeking the maximally probable sentence given an input MR; learning a translation model from

meaning to text. The inverse of WASP

Another example: Konstas and Lapata (2012). They use a set of grammar rules they defined to parse the database records and generate sentences according to the parsing results.

R(windSpeed) → *FS(temperature), R(rain)*: a description of *windSpeed* should be followed in the text by a temperature and a rain report.

Deep learning methods

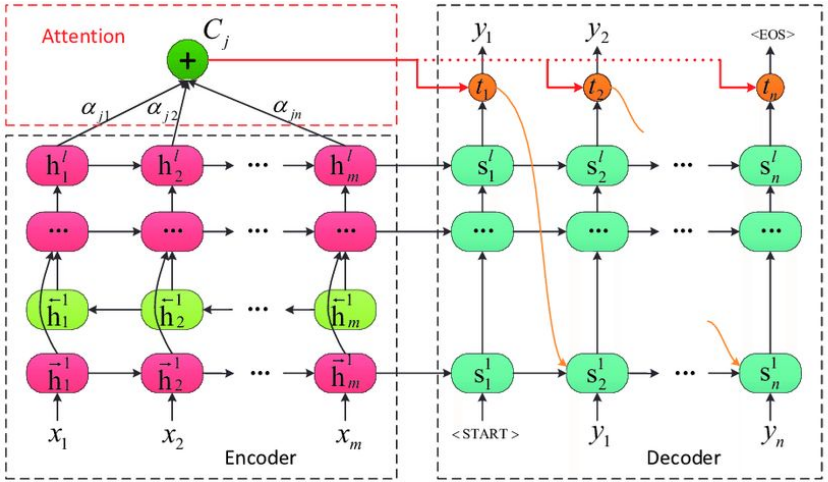
- Applications of deep neural network architectures
- Reasons:
  - 1. advances in hardware that can support resource-intensive learning problems
  - 2. NNs are designed to learn representations at increasing levels of abstraction by exploiting backpropagation
- Models:
  - feedforward networks,
  - log-bilinear models,
  - recurrent neural networks including **LSTM networks**
- Advantage:
  - handle sequences of varying lengths
  - avoiding both data sparseness and an explosion in the number of parameters

Encoder-Decoder Architecture

RNN is used to encode the input into a vector representation, which serves as the auxiliary input to a decoder RNN.

The use of attention mechanism forces the encoder to weight parts of the input during decoding

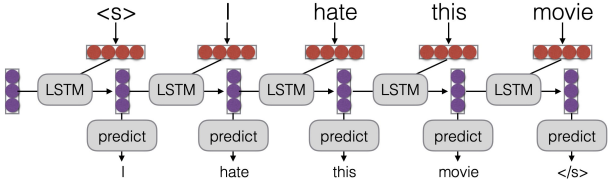
Application: Dialogue generation, machine translation



Conditioned Language Models

Tradition LM: $P(X) = \prod_{i=1}^{n} P(x_i | x_1, \ldots, x_{i-1})$

Next word — Context



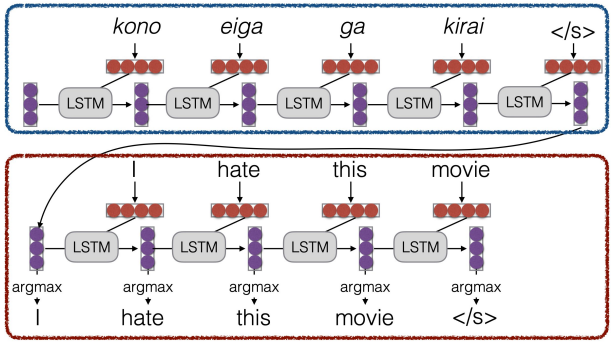
Conditioned LM: $P(Y|X) = \prod_{j=1}^{n} P(y_j | X, y_1, \ldots, y_{j-1})$

Added context

output is generated by sampling words or characters from a distribution conditioned on input feature

For different tasks, X represents different context

Input *X*	Output *Y* (**Text**)	Task
Structured Data	NL Description	NL Generation
English	Japanese	Translation
Document	Short Description	Summarization
Utterance	Response	Response Generation
Image	Text	Image Captioning
Speech	Transcript	Speech Recognition



Reference

Gatt A, Krahmer E. Survey of the state of the art in natural language generation: Core tasks, applications and evaluation[J]. Journal of Artificial Intelligence Research, 2018, 61: 65-170.

Mellish, C., Scott, D., Cahill, L., Paiva, D. S., Evans, R., & Reape, M. (2006). A Reference Architecture for Natural Language Generation Systems. Natural Language Engineering, 12(1), 1–34.

Reiter, E. (2007). An architecture for data-to-text systems. In Proc. ENLG'07, pp. 97–104.

Koller A, Stone M. Sentence generation as a planning problem[J]. 2007.

https://towardsdatascience.com/reinforcement-learning-demystified-markov-decision-processes-part-1-bf00dda41690

https://www.slideshare.net/YunNungVivianChen/deep-learning-for-dialogue-systems

Zhou, Long & Zhang, Jiajun & Zong, Chengqing. (2017). Look-ahead Attention for Generation in Neural Machine Translation.

http://phontron.com/class/nn4nlp2017/assets/slides/nn4nlp-08-condlm.pdf

Style Variation and Creative Text

Presented By Max Fowler

Outline

- What is “Style”? What is “Affect”?
- Different approaches to Style/Affect
- Creativity!
  - Jokes
  - Metaphors
  - Narrative
- Big Picture Takeaways and Comments

How do we define Style and Affect?

- Style -> the lexis, grammar, semantics that contribute to a text's context
  - e.g an author's style or the style of a medical report
  - The domain of "choice" - McDonald and Pustejovsky (1985)

- Affect -> The emotion reflect by a statement/words
  - Does an "um" mean someone is unsure or nervous?

Why care about Style and Affect?

- Match style to the audience and message
  - Don’t train medical robots on kid’s TV
  - “Hey there buddy, you’ve got cancer!”
- Match affect to the message and goal of the message
  - Uplifting: “Your donation today will help five foster puppies.”
  - Downer: “Without your donation, five puppies will starve.”

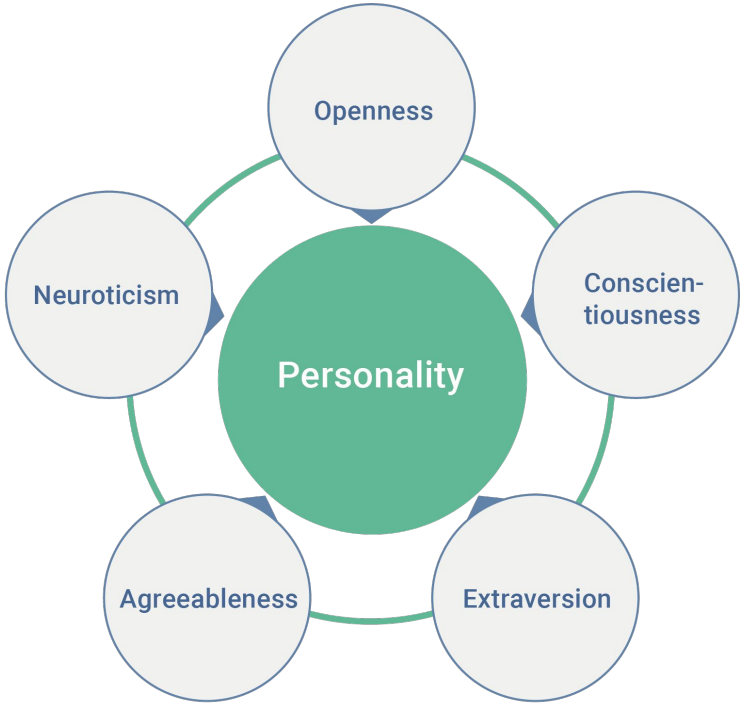
Rule-based Style

- Walker et al. 2002 SPOT planner
  - Boost correlations: sentence feature -> human perception
- Sheikha and Inkpen (2011) SimpleNLG extension

Sentence	Actual Class	Judge1 annotate	Judge2 annotate
The plane is going to leave on Jan. 5th.	Informal	Formal	Informal
They were transmuting the raw materials to finished goods.	Formal	Formal	Formal

Data-based Style

- A.K.A inductive view -> learn style from corpora features
- Hervas et al (2013) -> case-based, case on author
  - Model per author case - “Generate a Poe poem”
- PERSONAGE (2011) -> generate text using a goal AND Big 5 personality



PERSONAGE Example

- Goal - review Kin Khao and Tossed

1. Kin Khao and Tossed are bloody outstanding. Kin Khao just has rude staff. Tossed features sort of unmannered waiters, even if the food is somewhat quite adequate.

2. Err... I am not really sure. Tossed offers kind of decent food. Mmhm... However, Kin Khao, which has quite ad-ad-adequate food, is a thai place. You would probably enjoy these restaurants.

More about Affect

- Key agreement: “emotional states should impact lexical, syntactic, and other linguistic choices.”
- Empirical evaluation
  - Van der Sluis and Mellish (2010) -> Positive vs Neutral slant
  - “You crushed others on this test!” vs “You performed better than most students.”

Emotional Slant and Face

- Four strategies (Brown and Levinson 1987):
- Direct: Make my coffee!
- Approval: Would you mind making my coffee?
- Autonomy: Can you make the coffee?
- Indirect: Boy, I sure could use some coffee.

- Also face - positive (we share goals) and negative (don't get in my way)

Suggested Approach

- Largest focus is on response generation -> seq2seq, Encoder-Decoder trend
- Asghar et al. (2017) suggested approach
  a. Augment word embeddings with affect dictionary
  b. Decode with affect-sensitive beam search
  c. Train with an affect-sensitive loss function

Generating Creative Text

- Preciously little attention
- Gap between early creative AI and NLG

- Paper provides an overview of
  - Generating Puns and Jokes
  - Generating Metaphors
  - Generating Narratives

Why care about Creative Text Generation?

- “Good” writing holds attention - and creative text is part of that
- Expand computational creativity - can we make computers that are creative like people?
- Softball - add ML/AI assistance to traditional “creative” fields

The History of Atilla The Pun - Templates

- Joke Analysis and Production Engine (JAPE), Binsted and Ritchie, 1994-97
  - Template based NLG, “What do you call X?” e.g a “curious market”
  - Many lexical rules, such as juxtaposition
  - A: -> bizarre bazaars
- Petrovic and Matthews (2013) unsupervised templates
  - “I like my X like I like my Y, Z”
  - Laid out rules for funny jokes

Metaphor and Simile Generation

- All based on conceptual domain mapping
- Large focus on web data sets
  - Veale ‘07, ‘08, ‘13 - scraping and Google n-grams
- Hervas (2006) Narrative Context: “Luke Skywalker was the King Arthur of the Jedi Knights”

Most recent cited poetry - Zhang and Lapata (2017)

- Chinese Poetry Generation using RNNs

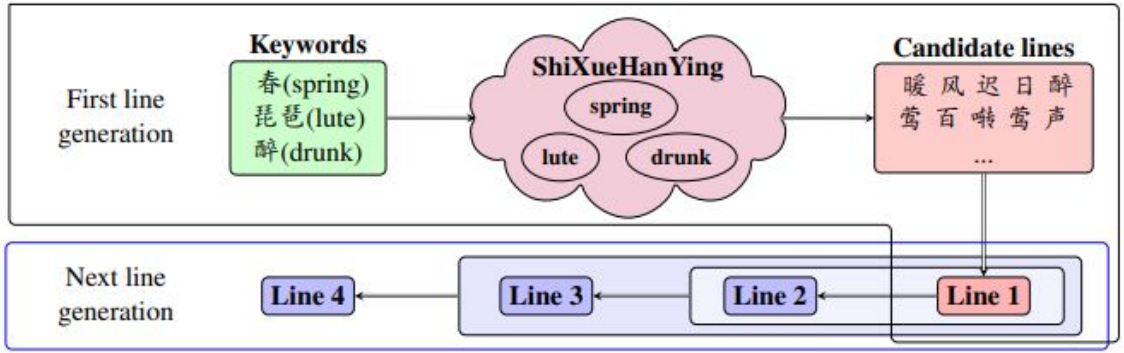


Figure 1: Poem generation with keywords *spring*, *lute*, and *drunk*. The keywords are expanded into phrases using a poetic taxonomy. Phrases are then used to generate the first line. Following lines are generated by taking into account the representations of all previously generated lines.

相 思

Missing You

红 豆 生 南 国， (*ZPPZ)

Red berries born in the warm southland.

春 来 发 几 **枝**？ (PPZZP)

How many branches flush in the spring?

愿 君 多 采 撷， (*PPZZ)

Take home an armful, for my sake,

此 物 最 相 **思**。 (*ZZPP)

As a symbol of our love.

Computational Narratology

- Branches from Formalist/Structuralist narratology: Bal 2009
- Narrative has:
  - Defining characteristics
  - Subtle features
- Difference between story and discourse
- In NLG: between text plan and the actual text

Pre-linguistic generation

- Generate plans within a story world (Gervas 2013 review)
- Example - TaleSpin problem solving vs generative Storybook

John Bear is somewhat hungry. John Bear wants to get some berries. John Bear wants to get near the blueberries. John Bear walks from a cave entrance to the bush by going through a pass through a valley through a meadow. John Bear takes the blueberries. John Bear eats the blueberries. The blueberries are gone. John Bear is not very hungry.

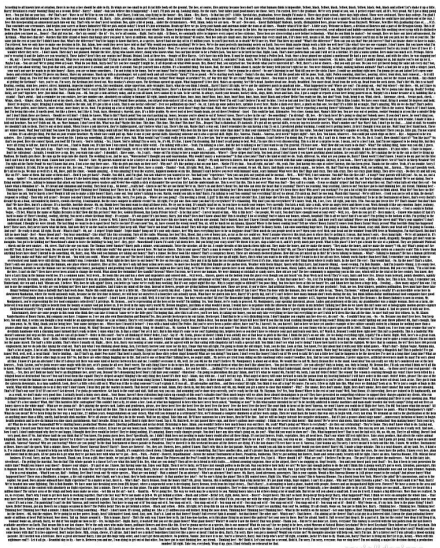
(a) Excerpt from TALESPIN

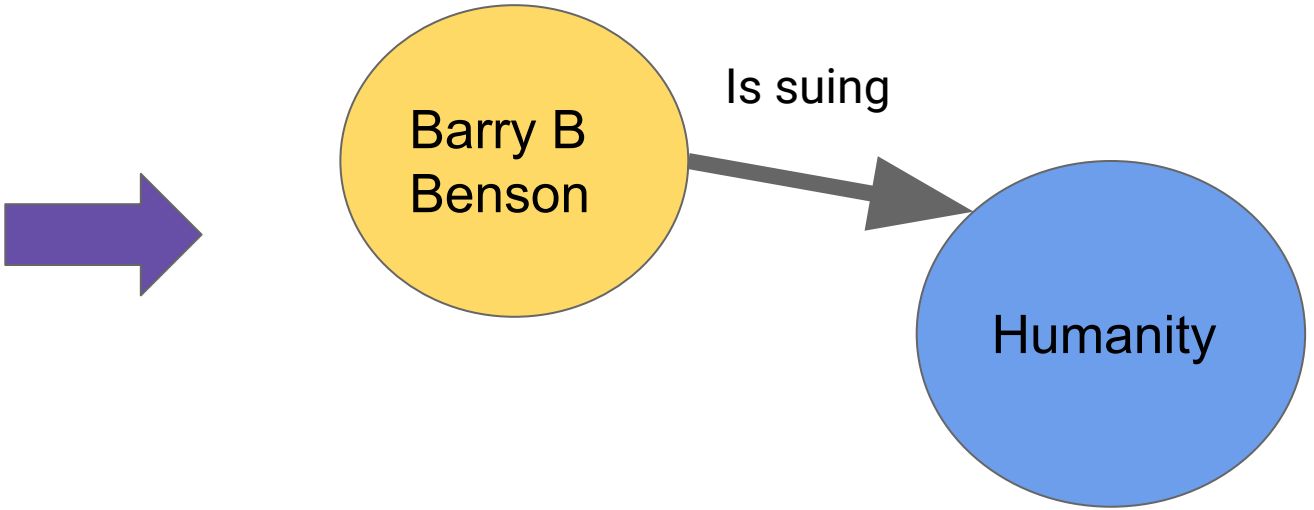
Once upon a time a woodman and his wife lived in a pretty cottage on the borders of a great forest. They had one little daughter, a sweet child, who was a favorite with every one. She was the joy of her mother's heart. To please her, the good woman made her a little scarlet cloak and hood. She looked so pretty in it that everybody called her Little Red Riding Hood.

(b) Excerpt from STORYBOOK

Actual data-driven narrative generation

- McIntyre and Lapata (2009) story generation -> database of entities and relations into a story





Actual data-driven narrative generation

- Most exciting - NaNoGenMon World Clock (Montfort 2013) 1440 (24 * 60) events

It is now exactly 05:00 in Samarkand. In some ramshackle dwelling a person who is called Gang, who is on the small side, reads an entirely made-up word on a box of breakfast cereal. He turns entirely around.

It is now right about 18:01 in Matamoros. In some dim yet decent structure a man named Tao, who is no larger or smaller than one would expect, reads a tiny numeric code from a recipe clipping. He smiles a tiny smile.

Takeaways from the paper

1. These forms of NLG => largely in infancy
2. Style and Affect lack clear agreement on “what” makes it and “how” they are perceived -> what conveys meaning and emotion?
3. How do we adapt to users in a live setting/in dialog?
4. NLG and old-style generative AI need to advance creative generation together
5. What is the evaluation metric? (More in Sec 7)

Critiques of this section

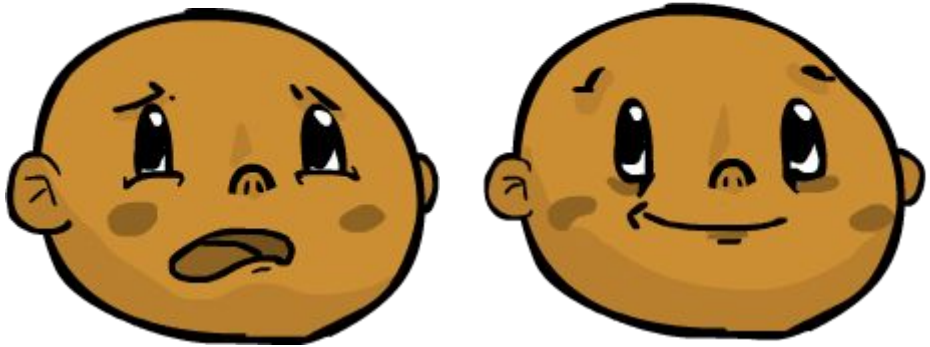
- Gleaning takeaways is “easy” and “hard”
  - Easy -> there was not a lot to talk about yet
  - Hard -> what WAS there was not well organized
- Organization of Style and Affect strange
  - Would like to see papers organized by model similarities
  - I will try to provide this, if feasible, in my write up

Non-paper Image Sources

Big-Five personality image from Wikipedia:

https://en.wikipedia.org/wiki/Big_Five_personality_traits#/media/File:Wiki-grafik_peats-de_big_five_ENG.png

Goofy little Potato Heads - commissioned by me years ago for a project, art by my former student Dylan Caldwell



Evaluation

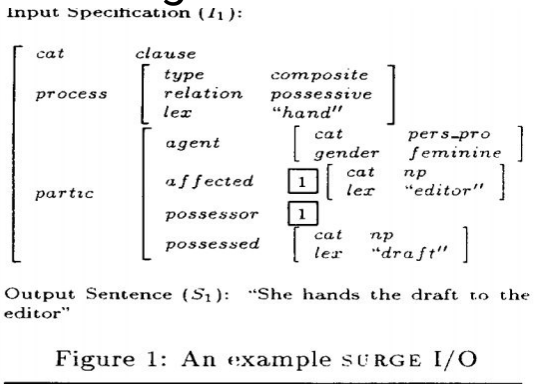
Presented By Ningkai Wu

System Evaluation is hard

- Variable input
- Variable outputs

System Evaluation is hard

- Variable input
  - No single, agreed-upon input format for NLG systems. One can only compare systems against a common benchmark if the input is similar, e.g. image-captioning systems.
  - For a common ‘standard’ dataset, comparison may not be straightforward due to input variation, e.g. fuf/surge realizer on the Penn Treebank.


Input Specification (I_1):

```
[ cat      clause
  process  [ type      composite  ]
           [ relation  possessive ]
           [ lex       "hand"     ]
  partic   [ agent     [ cat     pers_pro  ] ]
           [           [ gender  feminine  ] ]
           [ affected  [1] [ cat  np       ] ]
           [           [   [ lex  "editor" ] ]
           [ possessor [1] ]
           [ possessed [ cat  np       ] ]
           [           [ lex  "draft"  ] ]
```

Output Sentence (S_1): "She hands the draft to the editor"

Figure 1: An example SURGE I/O

System Evaluation is hard

- Variable outputs
  - Corpora often display a substantial range of variation and it is often unclear, without an independent assessment, which outputs are to be preferred (Reiter & Sripada, 2002).

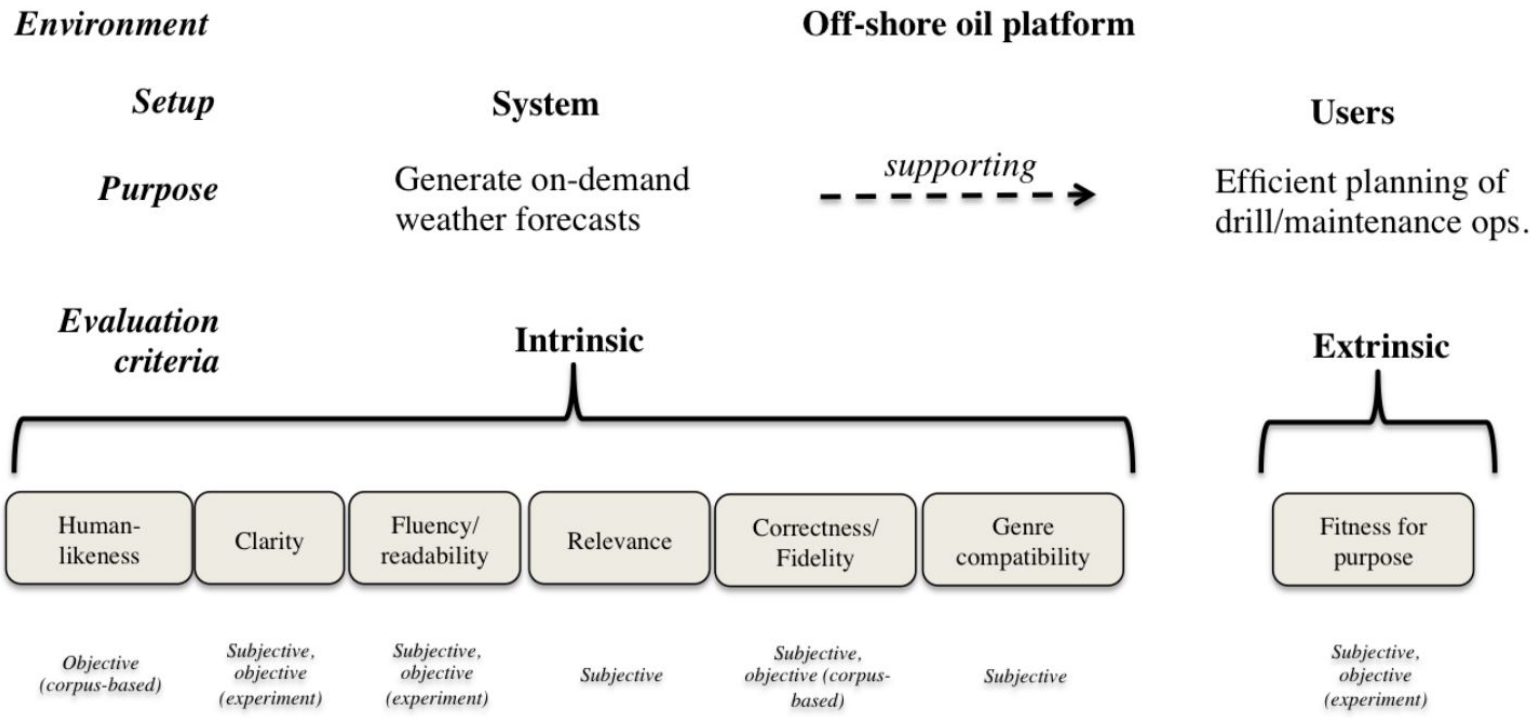
  - Capturing variation may itself be a goal , it is not always the case. E.g., SUMTIME-MOUSAM system weather forecasts were preferred by readers over those written by forecasters.

Scenario

A weather report generation system embedded in an offshore oil platform environment.

Goal: generate weather reports from numerical weather prediction data. Ultimately, facilitate users’ planning of drilling and maintenance operations.

Intrinsic and Extrinsic Evaluation Methods



Evaluation Methods: Intrinsic vs Extrinsic

An intrinsic evaluation measures the performance of a system without reference to other aspects of the setup, such as the system’s effectiveness in relation to its users.

Outline

- **Intrinsic Evaluation**
  - Human judgements based (subjective)
  - Corpora based

- Extrinsic Evaluation

- Relationship Between Evaluation Methods

Intrinsic Evaluation: human judgements

Human judgements are typically elicited by exposing naive or expert subjects to system outputs and getting them to rate them on some criteria. Common criteria include,

- Fluency or readability, that is, the linguistic quality of the text.
- Accuracy, adequacy, relevance or correctness relative to the input, reflecting the system's rendition of the content.

Intrinsic Evaluation: human judgements

Scale:

- Discrete, ordinal scales. (current dominant method)
- Continuous scales, e.g., a visually presented slider.

Intrinsic Evaluation: human judgements

More on scale: how do we help subjects find it easier to compare items rather than judge each one in its own right.

- Binary comparisons, e.g., the outputs of two mt systems
- Magnitude Estimation (Siddharthan and Katsos (2012)), e.g., subjects are not given a predefined scale, but are asked to choose their own and proceed to make comparisons of each item to a ‘modulus’, which serves as a comparison point.

Intrinsic Evaluation: human judgements

Inter-rater reliability:

Q: Multiple judgements by different evaluators may exhibit high variance.

A: Reduced by an iterative method whereby training of judges is followed by a period of discussion, leading to the updating of evaluation guidelines. (Godwin and Piwek(2016)).

Intrinsic Evaluation: Objective Humanlikeness Measures Using Corpora

Addressing the question of ‘humanlikeness’, that is, the extent to which the system’s output matches human output under comparable conditions.

- String overlap, string distance, or content overlap.
- Cheap, based on automatically computed metrics.

	Metric	Description	Origins
N-gram overlap	BLEU	Precision score over variable-length n-grams, with a length penalty (Papineni et al., 2002) and, optionally, smoothing (Lin & Och, 2004).	MT
	NIST	A version of BLEU with higher weighting for less frequent $n-$grams and a different length penalty (Doddington, 2002).	MT
	ROUGE	Recall-oriented score, with options for comparing non-contiguous $n-$grams and longest common subsequences (Lin & Hovy, 2003).	AS
	METEOR	Harmonic mean of unigram precision and recall, with options for handling (near-synonymy) and stemming (Lavie & Agarwal, 2007).	MT
	GTM	General Text Matcher. F-Score based on precision and recall, with greater weight for contiguous matching spans (Turian, Shen, & Melamed, 2003)	MT
	CIDEr	Cosine-based n-gram similarity score, with n-gram weighting using TF-IDF (Vedantam et al., 2015).	IC
	WMD	Word-Mover Distance, a similarity score between texts, based on the (semantic) distance between words in the texts (Kusner, Sun, Kolkin, & Weinberger, 2015). For NLP, distance is operationalised using normalised bag of words (NBOW) representations (Mikolov et al., 2013).	DS; IC
String distance	Edit distance	Number of insertions, deletions, substitutions and, possibly, transposition required to transform the candidate into the reference string (Levenshtein, 1966).	N/A
	TER	Translation edit rate, a version of edit distance (Snover, Dorr, Schwartz, Micciulla, & Makhoul, 2006).	MT
	TERP	Version of TER handling phrasal substitution, stemming and synonymy (Snover et al., 2006).	MT
	TERPA	Version of TER optimised for correlations with adequacy judgements (Snover et al., 2006).	MT
Content overlap	Dice/Jaccard	Set-theoretic measures of overlap between two unordered sets (e.g. of predicates or other content units)	N/A
	MASI	Measure of agreement between set-valued items, a weighted version of Jaccard (Passonneau, 2006)	AS
	PYRAMID	Overlap measure relying on comparison of weighted Summarization Content Units (SCUs) (Nenkova & Passonneau, 2004; Yang, Passonneau, & de Melo, 2016)	AS
	SPICE	Measure of overlap between candidate and reference texts based on propositional content obtained by parsing the text into graphs representing objects and relations, by first parsing captions into scene graphs representing objects and relations (Anderson, Fernando, Johnson, & Gould, 2016)	IC

BLEU Unigram Example

Candidate	the	the	the	the	the	the	the
Reference 1	the	cat	is	on	the	mat	
Reference 2	there	is	a	cat	on	the	mat

1. For each word in the candidate translation, take its maximum total count, m_max, in any of the reference translations. “The” appears once in ref1 and twice in ref2, thus m_max = 2.
2. For the candidate translation, the count m_w of each word is clipped to a maximum of m_max for that word. "the" has m_w = 7 and m_max = 2, thus m_w is clipped to 2.
3. Sum over m_w for each distinct words and then divide by the total number of unigrams in the candidate translation. Precision p_1 is 2/7 in this case.

$$\mathrm{BLEU} = \mathrm{BP} \cdot \exp\left(\sum_{n=1}^{N} w_n \log p_n\right)$$

Outline

- Intrinsic Evaluation
  - Human judgements based (subjective)
  - Corpora based

- Extrinsic Evaluation

- Relationship Between Evaluation Methods

Extrinsic Evaluation

In contrast to intrinsic methods, extrinsic evaluations measure effectiveness in achieving a desired goal. Dependent on the application domain and purpose of a system.

- Purchasing decision after presentation of arguments for and against options on the housing market based on a user model (Carenini & Moore, 2006);
- Persuasion and behaviour change, for example, through exposure to personalised smoking cessation letters (Reiter et al., 2003);

Extrinsic Evaluation

- Questionnaire-based or self-report.

- Objective measure of performance or achievement, e.g., GIVE Challenge (Striegnitz et al., 2011), in which NLG systems generated instructions for a user to navigate through a virtual world, a large-scale task-based evaluation was carried out by having users play the give game online.

Outline

- Intrinsic Evaluation
  - Human judgements based (subjective)
  - Corpora based
- Extrinsic Evaluation
- Relationship Between Evaluation Methods

Relationship Between Evaluation Methods

Weak correspondence between metrics and human judgements.

- Kulkarni et al. (2013)’s image description system preferred by human judgements but didn’t outperform on BLEU scores compared to other systems.
- Stent et al. (2005)’s paraphrase generation system, found that automatic metrics correlated highly with judgements of accuracy, but not fluency.

Possible Reasons

- Metrics such as BLEU are sensitive to the length of the texts under comparison. With shorter texts, n-gram based metrics are likely to result in lower scores.
- The type of overlap matters: for example, many evaluations in image captioning rely on BLEU-1, but longer n-grams are harder to match, though they capture more syntactic information and are arguably better indicators of fluency.
- Many intrinsic corpus-based metrics are designed to compare against multiple reference texts, but this is not always possible in NLG, e.g., image captioning datasets typically contain multiple captions per image.

Conclusion

- Conflicting results on the relationship between human judgements, behavioural measures and automatically computed metrics, depending on task and application domain.
- Use multiple evaluation methods in NLG to shed light on different aspects of quality.

Multi-domain Neural Network Language Generation for Spoken Dialog Systems

Presented By Samuel Kriman

Outline

1. Motivation
2. NLG Pipeline
3. Architecture
4. Training with Data Counterfeiting
5. Discriminative Objective Function
6. Datasets
7. Evaluation

Motivation

“Moving from limited-domain natural language generation (NLG) to open domain is difficult because the number of semantic input combinations grows exponentially with the number of domains. Therefore, it is important to leverage existing resources and exploit similarities between domains to facilitate domain adaptation.”

Proposed solution: train model on counterfeited data from an out-of-domain dataset, and then fine tuned on a small set of in-domain utterances with a discriminative objective function

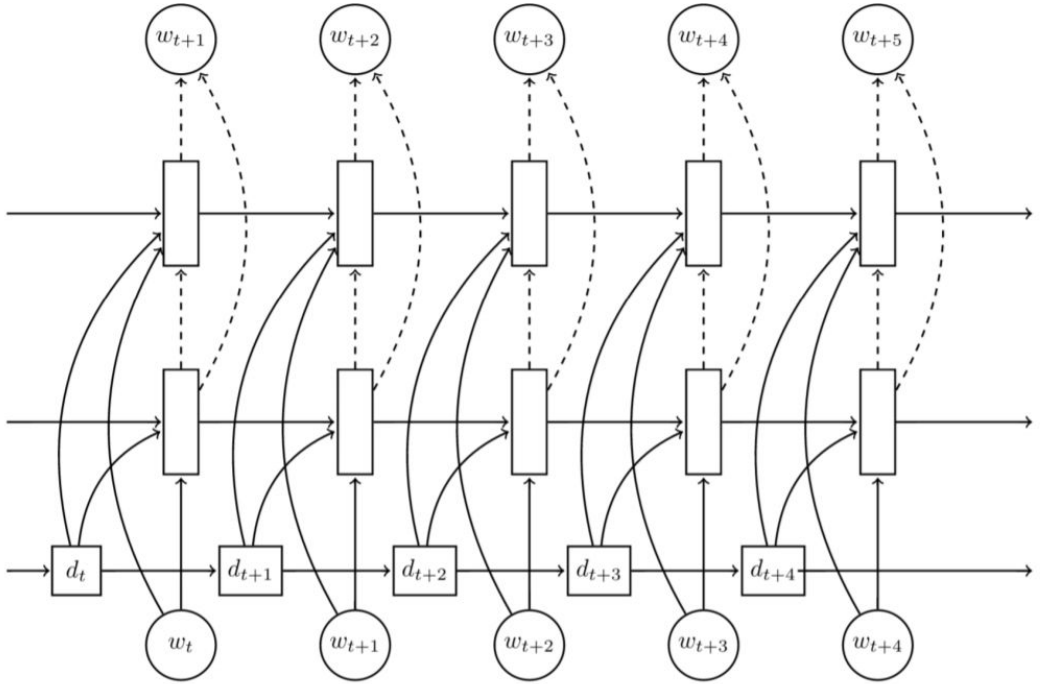
NLG Pipeline

Dialogue Act: A combination of an action type and a set of slot-value pairs.

Example: inform(name=”Seven days”,food=”chinese”)

Dialogue Act vector

Previously generated tokens



SC-LSTM

$$\begin{pmatrix} \mathbf{i}_t \\ \mathbf{f}_t \\ \mathbf{o}_t \\ \mathbf{r}_t \\ \hat{\mathbf{c}}_t \end{pmatrix} = \begin{pmatrix} \text{sigmoid} \\ \text{sigmoid} \\ \text{sigmoid} \\ \text{sigmoid} \\ \tanh \end{pmatrix} \mathbf{W}_{5n,2n} \begin{pmatrix} \mathbf{w}_t \\ \mathbf{h}_{t-1} \end{pmatrix}$$

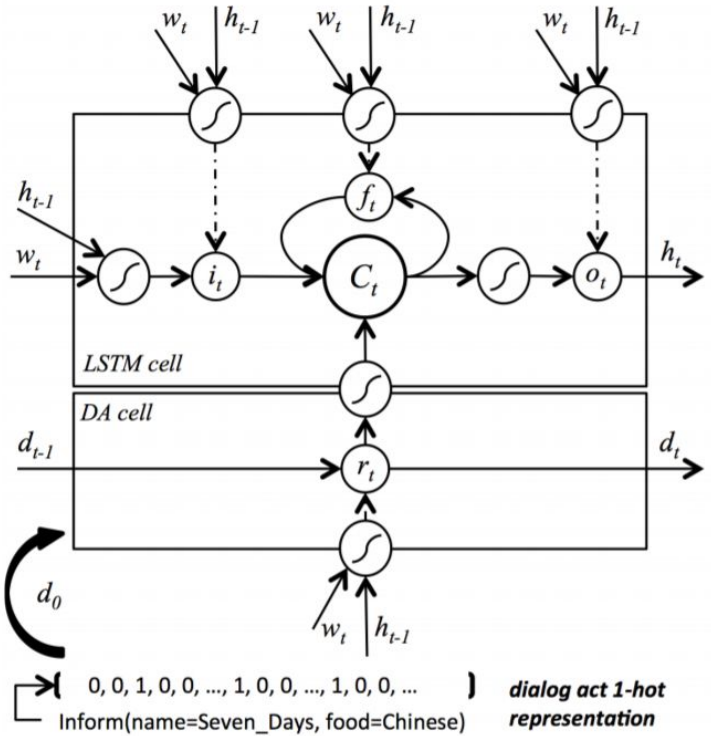
$$\mathbf{d}_t = \mathbf{r}_t \odot \mathbf{d}_{t-1}$$

$$\mathbf{c}_t = \mathbf{f}_t \odot \mathbf{c}_{t-1} + \mathbf{i}_t \odot \hat{\mathbf{c}}_t + tanh(\mathbf{W}_{dc}\mathbf{d}_t)$$

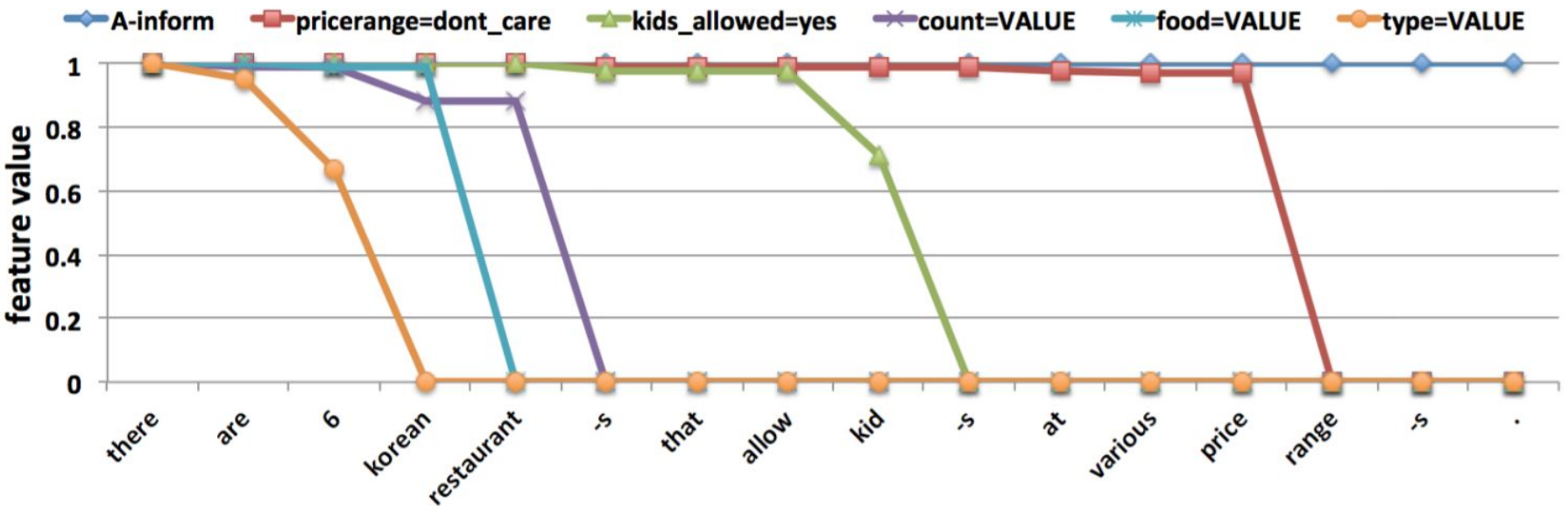
$$\mathbf{h}_t = \mathbf{o}_t \odot \tanh(\mathbf{c}_t)$$

$$p(w_{t+1}|w_t, w_{t-1}, ...w_0, \mathbf{d}_t) = softmax(\mathbf{W}_{ho}\mathbf{h}_t)$$

$$w_{t+1} \sim p(w_{t+1}|w_t, w_{t-1}, ...w_0, \mathbf{d}_t).$$



Dialogue Act vector propagation



Training with Data Counterfeiting

1. Categorise slots in both source and target domain into classes. In this case, they are separated based on their functional type into *informable*, *requestable*, and *binary*.
2. Delexicalise all slots and values
3. For each slot **s** in a source instance, randomly select a new slot that belongs to both the target ontology and the class of **s** to replace **s**. After replacing each slot in the instance we get a new pseudo-instance in the target domain.
4. Train a generator on the counterfeit dataset.
5. Refine parameters on real in-domain data.

Data counterfeiting example

An example realisation in laptop (source) domain:

Zeus 19 is a heavy laptop with a 500GB memory

delexicalisation ⇩

<R-NAME-value> is a <I-WEIGHT-value> <R-TYPE-value> with a <R-MEMEORY-value> <R-MEMORY-slot>

counterfeiting ⇩

<R-NAME-value> is a *<I-FAMILY-value> <R-TYPE-value> with a *<R-SCREEN-value> *<R-SCREEN-slot>

A possible realisation in TV (target) domain:

Apollo 73 is a U76 television with a 29-inch screen

Discriminative objective function

- Instead of maximising the log-likelihood of correct examples, DT aims at separating correct examples from competing incorrect examples
- We generate several candidates from a single DA and then use some scoring function L to compare them with ground truth

$$F(\theta) = -\mathbb{E}[L(\theta)]$$
$$= -\sum_{\Omega \in Gen(d_i)} p_\theta(\Omega|d_i) L(\Omega, \Omega_i)$$

$$p_\theta(\Omega|d_i) = \frac{\exp[\gamma \log p(\Omega|d_i,\theta)]}{\sum_{\Omega' \in Gen(d_i)} \exp[\gamma \log p(\Omega'|d_i,\theta)]}$$

$$\log p(\Omega|d_i,\theta) = \sum_{w_t \in \Omega} \log p(w_t|d_i,\theta)$$

Datasets

Datasets were used corresponding to 4 domains:

- Finding a restaurant
- Finding a hotel
- Buying a laptop
- Buying a television

The datasets were created by workers recruited by Amazon Mechanical Turk (AMT) by asking them to propose an appropriate natural language realisation corresponding to each system dialogue act actually generated by a dialogue system

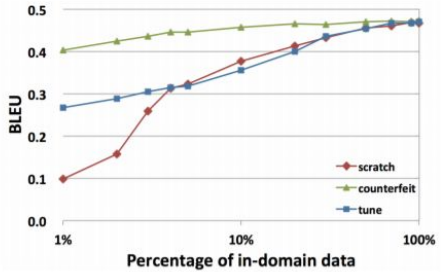
Datasets

In order to create more diverse datasets for the laptop and TV domains, the authors enumerated all possible combinations of dialogue act types and slots from the laptop and TV domains.

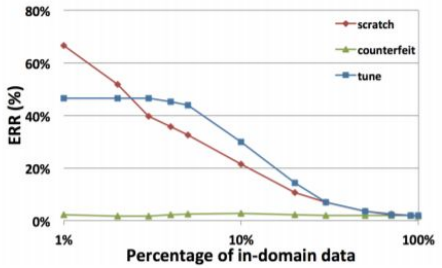
	Laptop	Television
informable slots	family, *pricerange, batteryrating, driverange, weightrange, **isforbusinesscomputing**	family, *pricerange, screensizerange, ecorating, hdmiport, **hasusbport**
requestable slots	*name, *type, *price, warranty, battery, design, dimension, utility, weight, platform, memory, drive, processor	*name, *type, *price, resolution, powerconsumption, accessories, color, screensize, audio
act type	*inform, *inform_only_match, *inform_on_match, inform_all, *inform_count, inform_no_info, *recommend, compare, *select, suggest, *confirm, *request, *request_more, *goodbye	

bold=binary slots, *=overlap with SF Restaurant and Hotel domains, all *informable slots* can take "dontcare" value

Automatic Evaluation

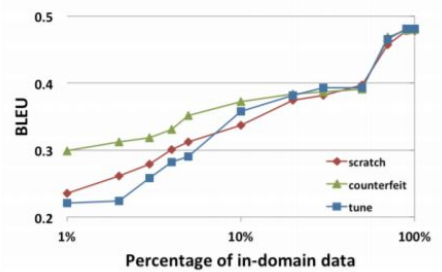


(a) BLEU score curve

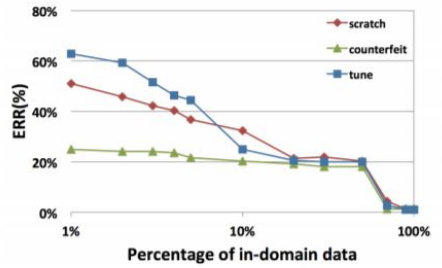


(b) Slot error rate curve

Figure 2: Results evaluated on TV domain by adapting models from laptop domain. Comparing train-from-scratch model (*scratch*) with model fine-tuning approach (*tune*) and data counterfeiting method (*counterfeit*). $10\% \approx 700$ examples.

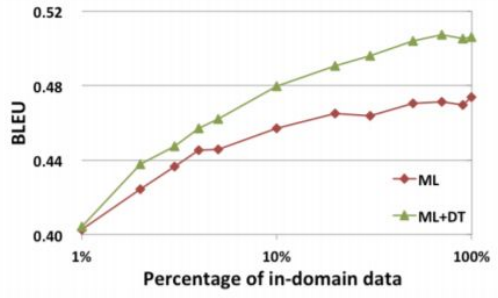


(a) BLEU score curve

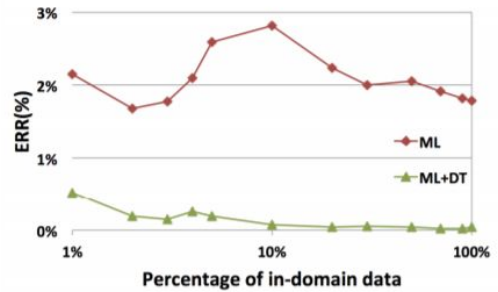


(b) Slot error rate curve

Figure 3: The same set of comparison as in Figure 2, but the results were evaluated by adapting from SF restaurant and hotel joint dataset to laptop and TV joint dataset. $10\% \approx 2K$ examples.



(a) Effect of DT on BLEU



(b) Effect of DT on slot error rate

Figure 4: Effect of applying DT training after ML adaptation. The results were evaluated on laptop to TV adaptation. $10\% \approx 700$ examples.

Human Evaluation

Method	TV to Laptop		laptop to TV	
	Info.	Nat.	Info.	Nat.
scrALL	2.64	2.37	2.54	2.36
DT-10%	**2.52****	**2.25****	**2.51**	2.19**
ML-10%	2.51**	2.22**	2.45**	**2.22****
scr-10%	2.24**	2.03**	2.00**	1.92**

* p <0.05, ** p <0.005

Table 2: Human evaluation for utterance quality in two domains. Results are shown in two metrics (rating out of 3). Statistical significance was computed using a two-tailed Student's t-test, between the model trained with full data (*scrALL*) and all others.

Pref.%	scr-10%	ML-10%	DT-10%	scrALL
scr-10%	-	34.5**	33.9**	22.4**
ML-10%	65.5**	-	44.9	36.8**
DT-10%	66.1**	55.1	-	35.9**
scrALL	77.6**	63.2**	64.1**	-

* p <0.05, ** p <0.005

(a) Preference test on TV to laptop adaptation scenario

Pref.%	scr-10%	ML-10%	DT-10%	scrALL
scr-10%	-	17.4**	14.2**	14.8**
ML-10%	82.6**	-	48.1	37.1**
DT-10%	85.8**	51.9	-	41.6*
scrALL	85.2**	62.9**	58.4*	-

* p <0.05, ** p <0.005

(b) Preference test on laptop to TV adaptation scenario

Conclusion

- The authors introduce a new procedure for training multi-domain, RNN-based language generators, by data counterfeiting and discriminative training
- Both automatic and human evaluation are performed, finding that good performance can be achieved with a small amount of in-domain data