

# Automatically responding to customers

February 12, 2019

# Table of Contents

---

- 1 Introduction
- 2 Preliminaries
- 3 Benchmarking
- 4 Improving accuracy
- 5 Conclusions

# Contents

---

- 1 Introduction
  - Research question 1
  - Research question 2
- 2 Preliminaries
- 3 Benchmarking
- 4 Improving accuracy
- 5 Conclusions

# Scope

---

Company goal was to automatically respond to customers using a few dozen pages of text.

Updated scope: Natural language understanding (NLU)

- Responding in real-time
- Few training examples

# Existing NLU benchmarks

---

- Braun et al. [1]
- Snips [3] (next slide)
- Burtsev et al. [2]
- Botfuel [5]

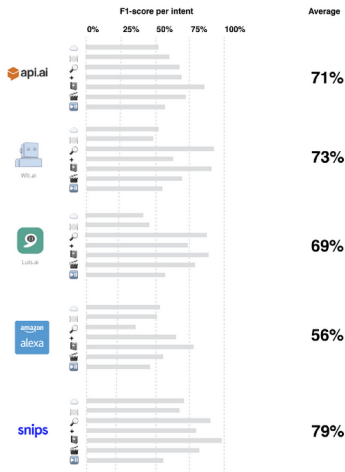
# Snips named-entity recognition (NER)

	I need a table in <span>Sacaton</span> at a <span>gluten free</span> restaurant	city	cuisine	✓
	I need a table in <span>Sacaton</span> at a <span>gluten</span> free restaurant	datetime	restaurant_type	✗
	I need a table in Sacaton at a gluten free restaurant			✗
	I need a table in Sacaton at a <span>gluten</span> free restaurant		restaurant_type	✗
	I need a table in <span>Sacaton</span> at a <span>gluten</span> free restaurant	country	restaurant_type	✗

Introduction  
Preliminaries  
Benchmarking  
Improving accuracy  
Conclusions

Research question 1  
Research question 2

# Their results



# Question and goal

---

- Can an open-source NLU benchmarking tool be created?
- Develop such a tool



# Question and goal

---

- Can the accuracy for NLU be increased?
- Improve the accuracy

# Contents

---

- 1 Introduction
- 2 Preliminaries
  - Natural language processing
  - Deep learning
- 3 Benchmarking
- 4 Improving accuracy
- 5 Conclusions

# Description of NLP field

---

For text or speech:

- Extract meaningful information
- Generate it

# Some well-known NLP tasks

---

- Machine translation
- Speech recognition

# Some well-known NLP tasks

- Machine translation
- Speech recognition
- **Named-entity recognition**

*What is [London's](**location**) weather [tomorrow](**date**)?*

# Some well-known NLP tasks

- Machine translation
- Speech recognition
- Named-entity recognition
- Intent classification

*What is [London's](location) weather [tomorrow](date)?*

*GetWeather*

# Language models

---

Try to model grammar

- Rule-based
- Statistical

# Language models

Try to model grammar

- Rule-based
- Statistical

Statistical applications:

Task	Example
Spell correction	$P(\text{my car broke}) > P(\text{my car boke})$
Machine translation	$P(\text{green house}) > P(\text{house green})$
Speech recognition	$P(\text{the red car}) > P(\text{she read ar})$





***Not tiger does that  
happy look***

***≠***

***That tiger does not  
look happy***

# Count-based

$$P(\text{the car broke}) = P(\text{the}) \cdot P(\text{car} \mid \text{the}) \cdot P(\text{broke} \mid \text{the car})$$

## Implementation last factor

$$P(\text{broke} \mid \text{the car}) = \text{COUNT}(\text{the car}) / \text{COUNT}(\text{broke})$$

## Approximation

$$P(\text{broke} \mid \text{the car}) \approx P(\text{broke} \mid \text{car})$$

# Count-based

$$P(\text{the car broke}) = P(\text{the}) \cdot P(\text{car} \mid \text{the}) \cdot P(\text{broke} \mid \text{the car})$$

## Implementation last factor

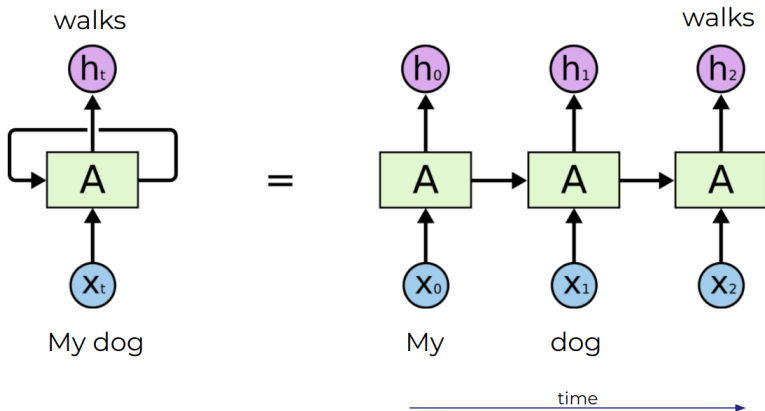
$$P(\text{broke} \mid \text{the car}) = \text{COUNT}(\text{the car}) / \text{COUNT}(\text{broke})$$

## Approximation

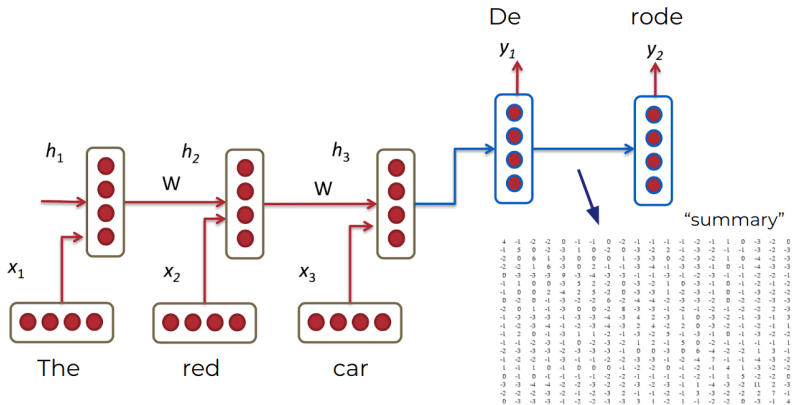
$$P(\text{broke} \mid \text{the car}) \approx P(\text{broke} \mid \text{car})$$

Insufficient history; not mobile-friendly

# Recurrent neural networks



# Translating



# Insufficient history

---

*Norwegian frigate sinking has far-reaching implications.*

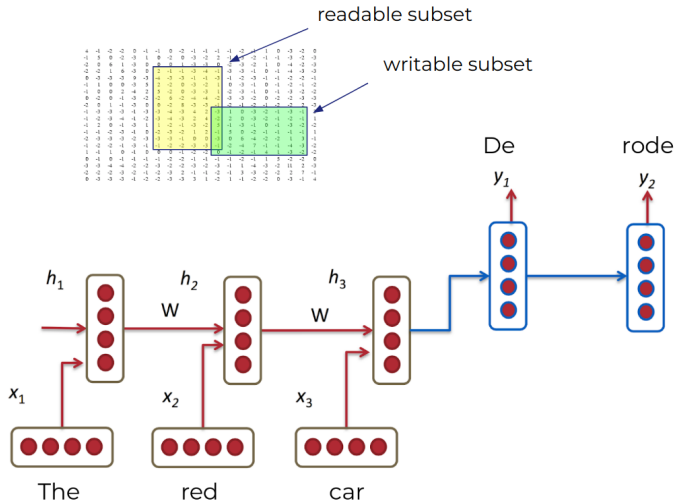
*Het zinken van het Noorse fregat heeft verstrekkende gevolgen.*

# Gated recurrent neural networks

---

- Gated recurrent unit (GRU)
- Long short-term memory (LSTM)

# Gates intuition





# Contents

---

- 1 Introduction
- 2 Preliminaries
- 3 Benchmarking**
  - Datasets
  - Systems
  - Tool and results
- 4 Improving accuracy
- 5 Conclusions

# Overview

<b>Dataset</b>	<b>Train</b>	<b>Test</b>	<b>Intents</b>	<b>Entities</b>
WebApplications	30	54	7	1
AskUbuntu	53	109	4	3
Chatbot	100	106	2	5
Snips2017	2100	700	7	unknown

## Example sentences

- WebApplications  
*How can I delete my [Hunch](WebService) account?*  
*DeleteAccount*
- Chatbot  
*when is the [next](criterion) [train](vehicle) in  
[muncher freiheit](StationStart)?*  
*DepartureTime*
- Snips2017  
*i want to listen to [Say it Again](track) by  
[Blackstratblues](artist)*  
*PlayMusic*

# Rasa

- open-source
- free
- local instance

The screenshot displays a chatbot interface with a purple border. The chat history shows a conversation with a customer named ACME. The customer asks, "Where should we send the confirmation to?". The chatbot responds, "Please send it to amy@example.com", accompanied by a female avatar icon. The customer then sends a thumbs-up emoji. Next, the customer asks, "Should we make that your primary email?". The chatbot responds, "Oh, what is it right now?", also with a female avatar icon. Below the chat history, a section titled "Next best action:" lists three actions with their respective confidence scores: "fetch\_primary\_email" (88%, checked), "hand\_off\_to\_human" (8%), and "fetch\_primary\_phone" (4%).

ACME: Where should we send the confirmation to?

Please send it to amy@example.com

👍

ACME: Should we make that your primary email?

Oh, what is it right now?

**Next best action:**

<input checked="" type="checkbox"/>	fetch_primary_email	88%
<input type="checkbox"/>	hand_off_to_human	8%
<input type="checkbox"/>	fetch_primary_phone	4%

Automatically responding to customers

# Rasa training data format

```
## intent:check_balance
- what is my balance <!-- no entity -->
- how much do I have on my [savings](source_account) <!-- entity "source_account" -->
- how much do I have on my [savings account](source_account:savings) <!-- synonym -->
- Could I pay in [yen](currency)? <!-- entity matched by lookup table -->

## intent:greet
- hey
- hello

## synonym:savings <!-- synonyms, method 2 -->
- pink pig

## regex:zipcode
- [0-9]{5}

## lookup:currencies <!-- lookup table list -->
- Yen
- USD
- Euro
```

# IBM Watson Conversation

[Skills](#) / [Customer Service - Sample](#) / Build

## Customer Service - Sample

A virtual assistant for customer service sample

**Intents**

Entities

Dialog

Content Catalog

Add intent



☐ Show only conflicts ?

<input type="checkbox"/> Intent (9) ▼	Description	Modified ▼	In Conflict	Examples
<input type="checkbox"/> #Cancel	Cancel the current request	5 months ago		7
<input type="checkbox"/> #Customer_Care_Appointments	Schedule or manage an in-st...	5 months ago		19
<input type="checkbox"/> #Customer_Care_Store_Hours	Find business hours.	5 months ago		38
<input type="checkbox"/> #Customer_Care_Store_Location	Locate a physical store locati...	5 months ago		23
<input type="checkbox"/> #General_Connect_to_Agent	Request a human agent.	5 months ago		47
<input type="checkbox"/> #General_Greetings	Greetings	5 months ago		30
<input type="checkbox"/> #Goodbye	Good byes	5 months ago		6
<input type="checkbox"/> #Help	Ask for help	5 months ago		6
<input type="checkbox"/> #Thanks	Thanks	5 months ago		8

Try it out

Clear

Manage Context 2

Hello, I'm a demo customer care virtual assistant to show you the basics. I can help with directions to my store, hours of operation and booking an in-store appointment

hi

#General\_Greetings ▼

Hello. Good morning

what are your opening hours?

#Customer\_Care\_Store\_Hours ▼

Our hours are Monday to Friday  
10am to 8pm and Friday and  
Saturday 11Am to 6pm.

Enter something to test your virtual assistant

Automatically responding to customers

# Tool: BENCH

- Python
- Docker containers
- API calls
- Not object-oriented<sup>1</sup>

---

<sup>1</sup>Steven Lott, Functional Python Programming

# Results

<b>System</b>	<b>Source</b>	<b>Ask- Ubuntu</b>	<b>Chatbot</b>	<b>Web- Apps</b>
Rasa:0.5-mitie	Braun et al.	0.862	0.981	0.746
Microsoft LUIS	Braun et al.	0.899	0.981	0.814
Watson	Braun et al.	0.917	0.972	0.831
Rasa:0.13.7-mitie	BENCH	0.881		0.763
Rasa:0.13.8-spacy	BENCH	0.853	0.981	0.627
Watson	BENCH	0.881	0.934	0.831
Dialogflow	BENCH	0.879	0.986	0.830



# Contents

---

- 1 Introduction
- 2 Preliminaries
- 3 Benchmarking
- 4 Improving accuracy**
  - BERT
  - Training
  - Joint training
  - Results

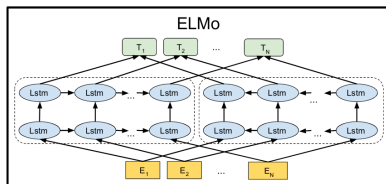
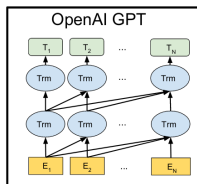
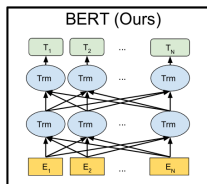
# Overview

---

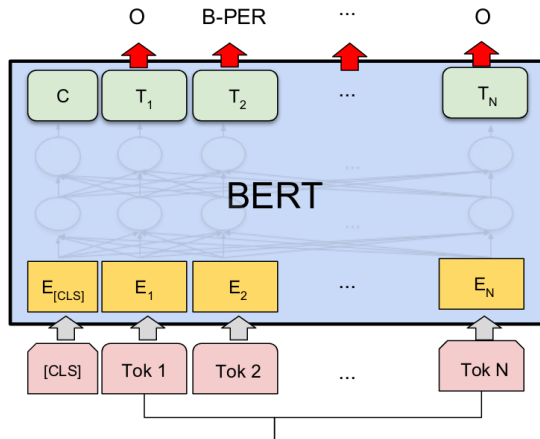
- December 2018
- SOTA 11 tasks
- Transformer (less sequential and  $\mathcal{O}(1)$  history)
- Pre-training
- Deep bidirectionality (next slide)

# Deep bidirectionality

*the ... on the hill*  
 $T_1 \quad T_2 \quad T_4 \quad T_5 \quad T_6$

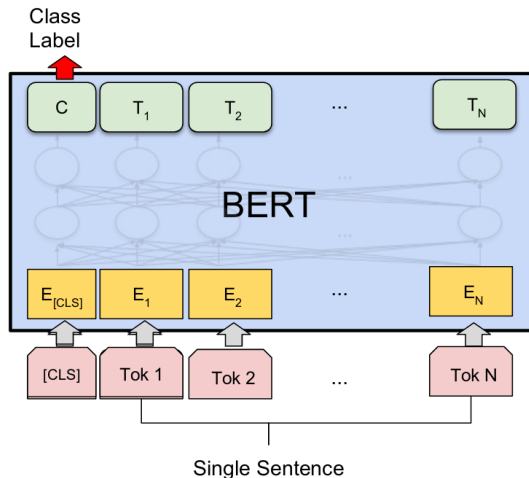


# Sequential labelling



● NER SOTA

# Single sentence classification



- Training time: Many hours on CPU
- Occasional near zero accuracy

# Intuition for intent classification

GetWeather:

*Will it rain in **London** **tomorrow**?*

*What is the **today's** temperature in **Madrid**?*

*Will it rain in **<location>** **<date>**?*

*What is the **<date>** temperature in **<location>**?*

# F<sub>1</sub> scores

Dataset	Steps	Method	Intent	Entity
Web-Apps	600 (twice)	Rasa separate	$0.67 \pm 0.04$ $0.72 \pm 0.03$	$0.81 \pm 0.01$
Ask-Ubuntu	600 (twice)	Rasa separate	$0.84 \pm 0.00$ $0.82 \pm 0.05$	$0.81 \pm 0.01$
Chatbot	600 (twice)	Rasa separate	$0.98 \pm 0.00$ $0.84 \pm 0.21$	$0.76 \pm 0.00$
Snips-2017	6000 (twice)	Rasa separate	$0.99 \pm 0.00$ $0.04 \pm 0.00$	$0.84 \pm 0.00$

# F<sub>1</sub> scores

Dataset	Steps	Method	Intent	Entity
Web-Apps	600 (twice) 600	Rasa	$0.67 \pm 0.04$	
		separate	$0.72 \pm 0.03$	$0.81 \pm 0.01$
		joint	$0.76 \pm 0.07$	$0.82 \pm 0.01$
Ask-Ubuntu	600 (twice) 600	Rasa	$0.84 \pm 0.00$	
		separate	$0.82 \pm 0.05$	$0.81 \pm 0.01$
		joint	$0.87 \pm 0.01$	$0.83 \pm 0.00$
Chatbot	600 (twice) 600	Rasa	$0.98 \pm 0.00$	
		separate	$0.84 \pm 0.21$	$0.76 \pm 0.00$
		joint	$0.98 \pm 0.00$	$0.79 \pm 0.00$
Snips-2017	6000 (twice) 6000	Rasa	$0.99 \pm 0.00$	
		separate	$0.04 \pm 0.00$	$0.84 \pm 0.00$
		joint	$0.98 \pm 0.02$	$0.86 \pm 0.00$



## Future work

---

- Code validation
- Loss function
- Entities baseline comparison
- More datasets
- Evaluate newer architectures, such as evolved ('mobile-friendly') transformer [4]

# Contents

---

- 1 Introduction
- 2 Preliminaries
- 3 Benchmarking
- 4 Improving accuracy
- 5 Conclusions
  - Research question 1
  - Research question 2

# Can an open-source NLU benchmarking tool be created?

Yes. Requirements:

- Continuous maintenance
- Support vendor APIs
- More metrics
- Multiple runs
- More datasets

# Can the accuracy for NLU be increased?

---

Yes. Each few months a new SOTA paper.

Why BERT is suspected to have improved SOTA:

- SOTA NER
- Deeply bidirectional
- More history.

Further work: Whether accuracy improvements are significant.

# References I

Braun, D., Hernandez-Mendez, A., Matthes, F., & Langen, M. (2017). Evaluating natural language understanding services for conversational question answering systems. In *Proceedings of the 18th annual SIGdial meeting on discourse and dialogue* (pp. 174–185).

Burtsev, M., Seliverstov, A., Airapetyan, R., Arkhipov, M., Baymurzina, D., Bushkov, N., ... others (2018).

DeepPavlov: Open-source library for dialogue systems. *Proceedings of ACL 2018, System Demonstrations*, 122–127.

## References II

Coucke, A. (2017). *Benchmarking natural language understanding systems: Google, Facebook, Microsoft, Amazon, and Snips*.

<https://medium.com/snips-ai/2b8ddcf9fb19>. (Accessed: 2019-01-18)

So, D. R., Liang, C., & Le, Q. V. (2019). The evolved transformer. *arXiv preprint arXiv:1901.11117*.

Trong Canh, N. (2018). *Benchmarking intent classification services - June 2018*.

<https://medium.com/botfuel/eb8684a1e55f>. (Accessed: 2019-01-18)