



Recent Developments in Spoken Language Understanding

Henry Weld, Siqu Long, Josiah Poon, Soyeon Caren Han

***ADNLP (Australia Deep Learning Natural Language Processing Team)
University of Sydney and University of Western Australia***

AJCAI 2022, Perth

An abstract background featuring overlapping, semi-transparent teal-colored geometric shapes, primarily triangles and polygons, creating a complex, crystalline pattern that tapers off towards the right side of the slide.

Organisers and Presenters

[Dr. Henry Weld](#), The University of Sydney

[Ms. Siqu Long \(Sharon\)](#), The University of Sydney

[Dr. Josiah Poon](#), The University of Sydney

[Dr. Caren Han](#), University of Western Australia and The University of Sydney

An abstract background composed of overlapping, semi-transparent teal-colored polygons of various shapes and sizes, creating a complex, crystalline structure that fills the upper half of the slide.

Publications and Expertise

- Weld, H., Huang, X., Long, S., Poon, J., Han, S.C. (2022). A survey of joint intent detection and slot-filling models in natural language understanding. **ACM Computing Surveys**.
- Weld, H., Huang, G., Lee, J., Zhang, T., Wang, K., Guo, X., Long, S., Poon, J., Han, S.C. (2021, August). CONDA: a CONtextual Dual-Annotated dataset for in-game toxicity understanding and detection. ACL-IJCNLP, Bangkok, Thailand (pp. 2406–2416). Association for Computational Linguistics, **ACL 2021**
- Han, S.C., Long, S., Li, H., Weld, H., Poon, J. (2021, August). Bi-Directional Joint Neural Networks for Intent Classification and Slot Filling. Proc. Interspeech. **Interspeech 2021**, Brno, Czechia (pp. 4743-4747). ISCA. doi: 10.21437/Interspeech.2021-2044

An abstract background composed of overlapping, semi-transparent teal-colored polygons of various shapes and sizes, creating a complex, crystalline structure that fills the upper half of the slide.

Today's plan

Time	Topic
13:30 - 14:00	Introduction to Natural Language Processing (NLP) and Spoken Language Understanding (SLU)
14:00 - 14:30	Joint SLU Approaches
14:30 - 14:45	QnA and Break
14:45 - 15:00	Hands-on Exercise (Joint BERT)
15:00 - 15:15	SLU Evaluation
15:15 - 16:00	Hands-on Exercise (Datasets, Metrics, Experiments)
16:00 - 16:15	Future Direction
16:15 - 16:30	QnA

The background of the slide features an abstract, low-poly geometric design in various shades of teal and light blue. The shapes are interconnected, creating a sense of depth and movement, particularly on the right side where the polygons are more complex and layered.

Introduction to NLP and SLU

13:30 - 14:00



Natural Language Processing

The interface between computers and language

Natural Language Understanding

Natural Language Generation

Question Answering and Information Retrieval

Translation

Summarisation

Sentiment analysis

Syntactic structure and component detection

Dependency and constituency parsing (e.g. part of speech)

Named entity recognition

Coreference resolution

Conversation

Spoken Dialogue System



Google Home



Amazon Alexa



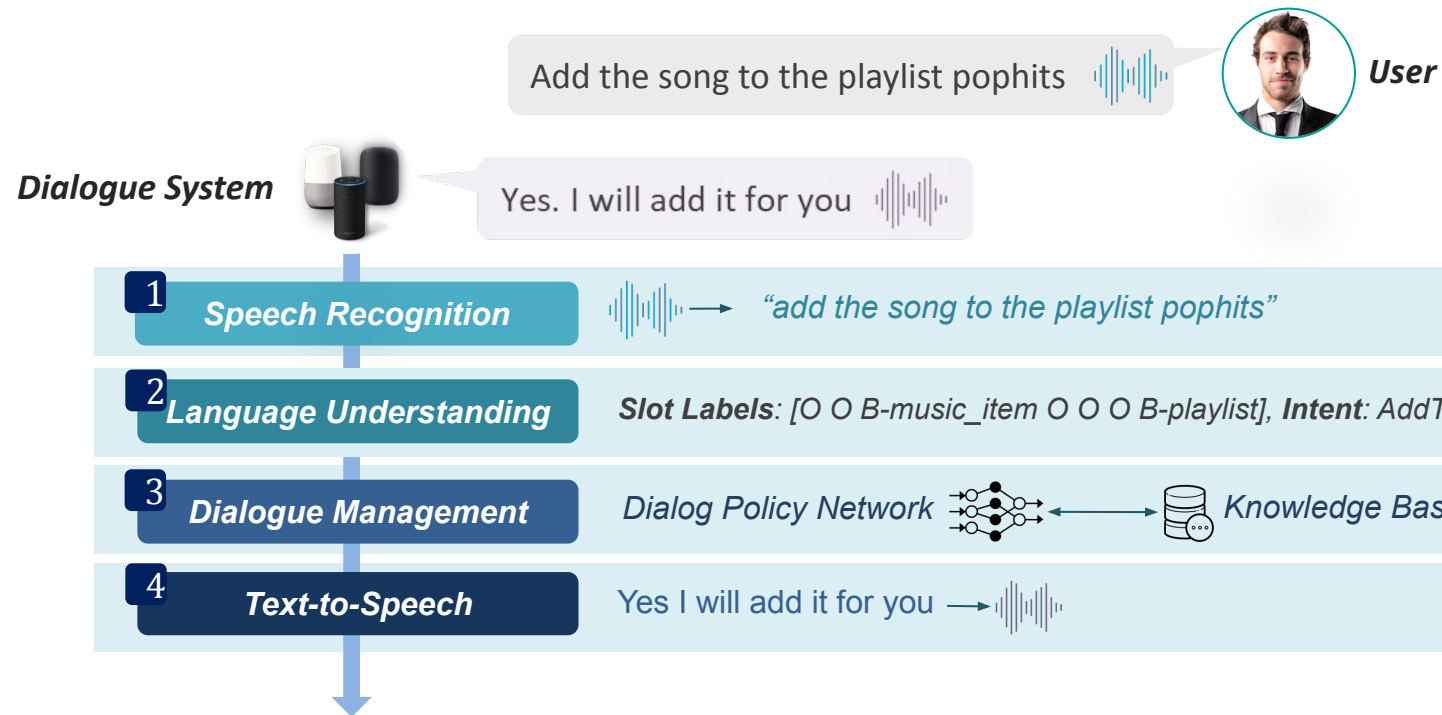
Apple Siri HomePod

“The Successful Spoken Dialogue Model is the key component in today’s virtual personal assistants. (Sundar Pichai, Google CEO)”

→ *Allow users to speak naturally to finish the task*

Spoken Dialogue System

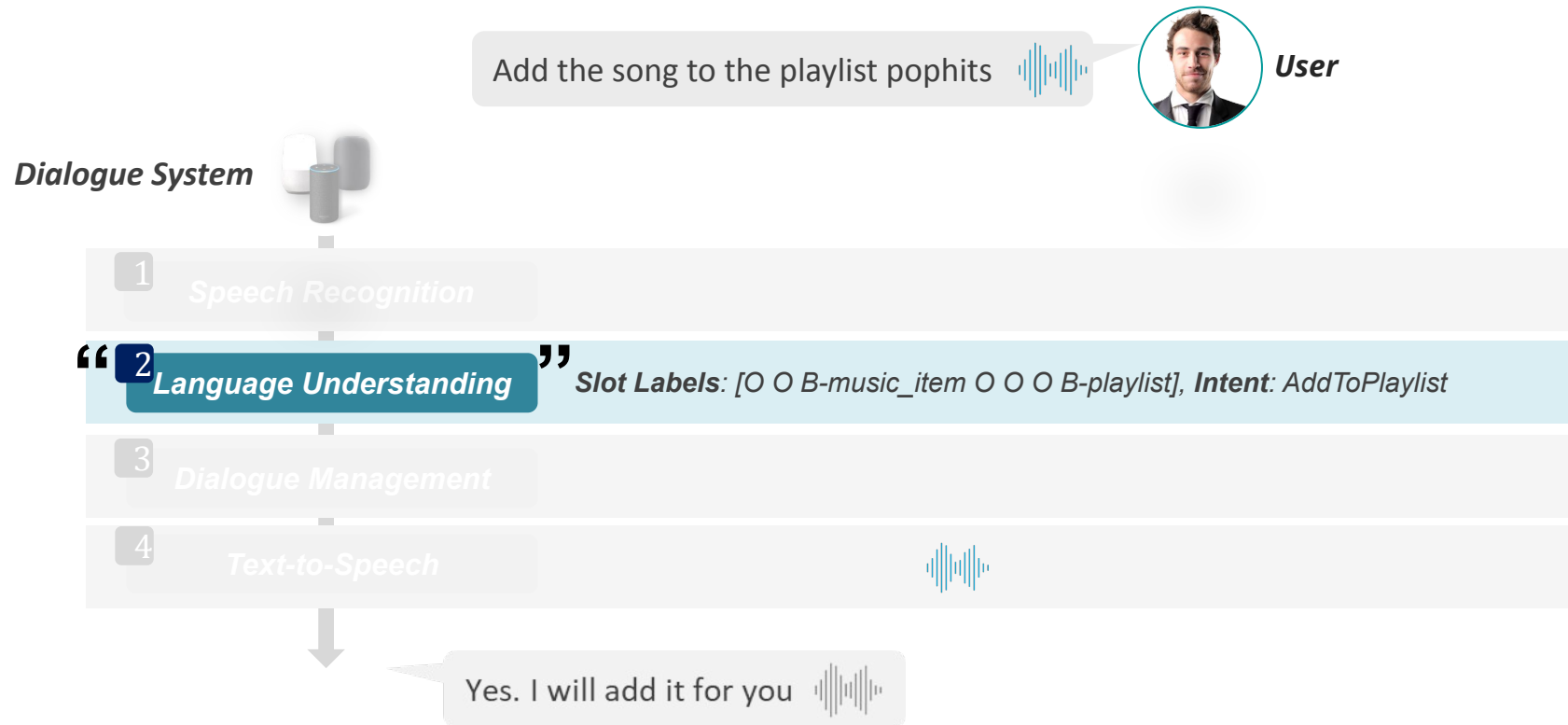
The Successful Spoken Dialogue Model



Spoken Dialogue System

The Successful Spoken Dialogue Model

To build the successful spoken dialogue model, Natural Language Understanding is the first crucial step to pass



Natural Language Understanding (NLU)

Natural Language Understanding Tasks

To build the successful spoken dialogue model, Natural Language Understanding is the first crucial step to pass *but considered as an AI-hard problem*

Add the song to the playlist pophits



User

Language Understanding Tasks

1) Interprets the semantic context (Slot Labels) that the user communicates and 2) classifies it into proper intents

Utterance

Add this song to the playlist pophits

Task 1) Slot Filling

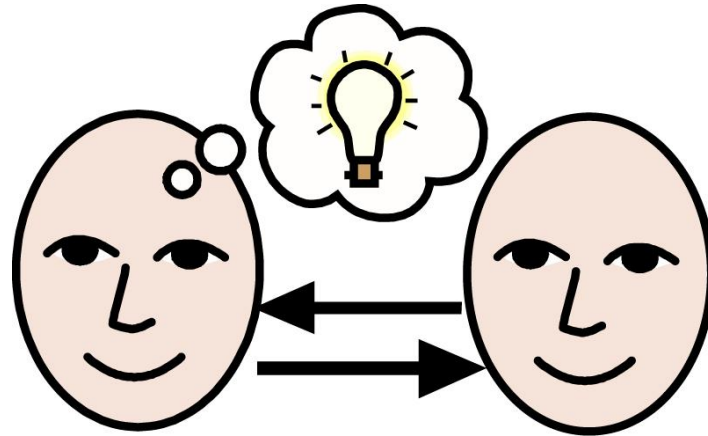
O O B-musitem O O O B-playlist

Task 2) Intent Classification

AddToPlaylist

Natural Language UNDERSTANDING

What do we mean by understanding?



Natural Language UNDERSTANDING

What do we mean by understanding?

Understand: perceive the intended meaning of (words, a language, or a speaker), Oxford dictionary

Understanding is a psychological process related to an abstract or physical object, such as a person, situation, or **message** whereby one is able to **use concepts to model that object**. Understanding **implies abilities and dispositions** with respect to an object of knowledge that are sufficient **to support intelligent behaviour**. Carl Bereiter, Education and Mind In The Knowledge Age, 2006

Semantic frame

Domain - what area is the speaker talking about?

Intent - what is the intent of each their utterances?

Slots - which words in the utterance carry the semantic information of the details of the intent, and what semantic category do they represent?

Semantic frame example - goal oriented chatbot

So from there, you- again you can take the subway to the Clarke Quay.

From HarbourFront to Clarke Quay.

I think- Oh.

Okay, there are quite a few restaurants there.

%Uh and you can have your chili crabs out there.

Okay, great.

%Um what about for %uh you know, shopping.

%Uh because I think one %um Singapore is also %um popular with- when it comes to- because you know, we're about to go home already, right?

Because we're done with dinner and we have- before we leave, so we have to shop.

So %um I know that Singapore is very popular when it comes to gadgets.

Right.

Yes.

Right.

%Um because I've heard from my friends that you can actually buy like gadgets in Singapore %uh for a much cheaper price.

And also the quality is really good.

So %um do you know for a place wherein, you know, my husband because he's a techy person,

Right.

my husband can go and maybe, you know, check for cell phones, laptop or iPad, something like that?

Alright, you can go to this place.

%Uh again if you're going to stay in the Lavender MRT area.

%Uh it's just like two station away from where you are.

Source: DSTC4 dataset

Semantic frame example - goal oriented chatbot

So from there, you- again you can take the subway to the Clarke Quay.
From HarbourFront to Clarke Quay.
I think- Oh.

Domain: Transportation

Okay, there are quite a few restaurants there.
%Uh and you can have your chili crabs out there.

Domain: Food

Okay, great.

%Um what about for %uh you know, shopping.

**%Uh because I think one %um Singapore is also %um popular with- when it comes to-
because you know, we're about to go home already, right?**

Because we're done with dinner and we have- before we leave, so we have to shop.

So %um I know that Singapore is very popular when it comes to gadgets.

Right.

Yes.

Right.

%Um because I've heard from my friends that you can actually buy like gadgets in Singapore %uh for a much cheaper price.

And also the quality is really good.

So %um do you know for a place wherein, you know, my husband because he's a techy person,

Right.

my husband can go and maybe, you know, check for cell phones, laptop or iPad, something like that?

Alright, you can go to this place.

%Uh again if you're going to stay in the Lavender MRT area.

%Uh it's just like two station away from where you are.

Domain: Shopping

Source: DSTC4 dataset

Semantic frame example - goal oriented chatbot

So from there, you- again you can take the subway to the Clarke Quay. **FOL-HOW_TO**

From HarbourFront to Clarke Quay. **FOL-HOW_TO**

I think- Oh. **FOL-EXPLAIN**

Domain: Transportation

Okay, there are quite a few restaurants there. **FOL-INFO**

%Uh and you can have your chili crabs out there. **FOL-RECOMMEND**

Domain: Food

Okay, great. **FOL-POSITIVE**

Domain: Shopping

%Um what about for %uh you know, shopping. **INI-EXPLAIN**

%Uh because I think one %um Singapore is also %um popular with- when it comes to-

because you know, we're about to go home already, right? **FOL-EXPLAIN**

Because we're done with dinner and we have- before we leave, so we have to shop. **FOL-EXPLAIN**

So %um I know that Singapore is very popular when it comes to gadgets. **FOL-EXPLAIN**

Right. **FOL-CONFIRM**

Yes. **FOL-ACK**

Right. **FOL-ACK**

%Um because I've heard from my friends that you can actually buy like gadgets in Singapore %uh for a much cheaper price. **FOL-EXPLAIN**

And also the quality is really good. **FOL-EXPLAIN**

So %um do you know for a place wherein, you know, my husband because he's a techy person, **FOL-RECOMMEND-WHERE**

Right. **FOL-ACK**

my husband can go and maybe, you know, check for cell phones, laptop or iPad, something like that? **FOL-EXPLAIN-PREFERENCE**

Alright, you can go to this place. **FOL-RECOMMEND-WHERE**

%Uh again if you're going to stay in the Lavender MRT area. **FOL-EXPLAIN-WHERE**

%Uh it's just like two station away from where you are. **FOL-WHERE**

Source: DSTC4 dataset

Semantic frame example - goal oriented chatbot

So from there, you- again you can take the subway to the Clarke Quay. **FOL-HOW_TO**

From HarbourFront to Clarke Quay. **FOL-HOW_TO**

I think- Oh. **FOL-EXPLAIN**

Domain: Transportation

Okay, there are quite a few restaurants there. **FOL-INFO**

%Uh and you can have your chili crabs out there. **FOL-RECOMMEND**

Domain: Food

Okay, great. **FOL-POSITIVE**

Domain: Shopping

%Um what about for %uh you know, shopping. **INI-EXPLAIN**

%Uh because I think one %um Singapore is also %um popular with- when it comes to-

because you know, we're about to go home already, right? **FOL-EXPLAIN**

Because we're done with dinner and we have- before we leave, so we have to shop. **FOL-EXPLAIN**

So %um I know that Singapore is very popular when it comes to gadgets. **FOL-EXPLAIN**

Right. **FOL-CONFIRM**

Yes. **FOL-ACK**

Right. **FOL-ACK**

%Um because I've heard from my friends that you can actually buy like gadgets in Singapore %uh for a much cheaper price. **FOL-EXPLAIN**

And also the quality is really good. **FOL-EXPLAIN**

So %um do you know for a place wherein, you know, my husband because he's a techy person, **FOL-RECOMMEND-WHERE**

Right. **FOL-ACK**

my husband can go and maybe, you know, check for cell phones, laptop or iPad, something like that? **FOL-EXPLAIN-PREFERENCE**

Alright, you can go to this place. **FOL-RECOMMEND-WHERE**

%Uh again if you're going to stay in the Lavender MRT area. **FOL-EXPLAIN-WHERE**

%Uh it's just like two station away from where you are. **FOL-WHERE**

Source: DSTC4 dataset

Applications

Question answering, information gathering, search

Bing

How far from Perth to Bunbury?

- Q how far **is bunbury** from perth
- Q how far **bunbury** from perth
- Q how far perth to **bunbury**
- Q how far from perth to **bunbury**

Google

Q How far from Perth to Bunbury

- G How far from Perth to Bunbury — Search with Google
- Q **is there a train** from perth to bunbury
- Q **can you catch a train** from perth to bunbury

Applications

Instruction - to robots, driverless vehicles, IoT, personal assistants

Conversational agents

- Task oriented (limited domain)
- Chatbots or open domain

What about implicit intents?

- Sarcasm, humour
- Power plays, hate speech, toxicity

Third party monitoring



Basic dual level semantic frame

Intent - a labelling of the sentence with the speaker's intent, from a finite set of classes

"i would like to find a flight from charlotte to las vegas that makes a stop in st. louis"

"i want to fly from boston at 830 am and arrive in denver around 11 in the morning"

The intent is **find_flight**

"add purple rain to playlist night songs"

The intent is **add_to_playlist**

Intent classification

To identify the intent we perform a classification task:

- classification is a mapping from the input object to one of the members of a finite set of classes

Basic dual level semantic frame

Slots - a labelling of each token in the sentence with its semantic role

Example:

"i would like to find a flight from charlotte to las vegas that makes a stop in st. louis"

Basic dual level semantic frame

Slots - a labelling of each token in the sentence with its semantic role

Example:

"i would like to find a flight from charlotte to las vegas that makes a stop in st. louis"

"i would like to find a flight from **charlotte** to **las vegas** that makes a stop in **st. louis**"
city from city to stopover city

Tokenisation - an aside

Tokenisation in NLP is breaking a text up into its constituent units for analysis.

A **corpus** is a collection of texts

A text may be broken into chapters, paragraphs, sentences, words, alphanumeric characters, encoded characters, n-grams.

e.g. bigrams

“word” -> [“wo”, “or”, “rd”]

We may perform certain standardising operations on the tokens (e.g. convert to all lower case, remove punctuation) depending on the application we are working on, this is called **pre-processing**

Slot filling

The slots in NLU are labelled using BIO tagging.

B for beginning

I for inside

O for outside, or Other

Chunks or spans. Multi-word expressions.

"i would like to find a flight from charlotte to las vegas that makes a stop in st. louis"

O O O O O O O O B-fromloc.city O B-toloc.city I-toloc.city O O O O O B-stoploc.city I-stoploc.city

Slot filling

Slot filling is a sequence labelling task, given the tokens identify the label for each from a pre-defined, finite set of labels.

"i would like to find a flight from charlotte to las vegas that makes a stop in st. louis"

○ ○ ○ ○ ○ ○ ○ ○ B-fromloc.city ○ B-toloc.city I-toloc.city ○ ○ ○ ○ ○ B-stoploc.city I-stoploc.city

ATIS

0: flight: BOS i want to fly from boston at 838 am and arrive in denver at 1110 in the morning EOS

BOS	O
i	O
want	O
to	O
fly	O
from	O
boston	B-fromloc.city_name
at	O
838	B-depart_time.time
am	I-depart_time.time
and	O
arrive	O
in	O
denver	B-toloc.city_name
at	O
1110	B-arrive_time.time
in	O
the	O
morning	B-arrive_time.period_of_day
EOS	O

SNIPS-NLU

```
"intent": {  
  "GetWeather": {  
    "utterance": [  
      {  
        "data": [  
          {  
            "text": "give me the weather forecast for "  
          },  
          {  
            "text": "los angeles",  
            "entity": "location",  
            "slot_name": "weatherLocation"  
          },  
          {  
            "text": "this weekend",  
            "entity": "snips/datetime",  
            "slot_name": "weatherDate"  
          }  
        ]  
      }  
    ],  
    ...
```

Machine learning issues with sequence labelling

Sequence labelling

- label dependency - capturing the prior distribution of label co-occurrence
- long range label dependency
- label bias

Intuitively, how do you understand?

Intuitively, how do you understand?



priyankadutta.com

Joint Intent Detection and Slot filling

The two tasks should actively inform each other ...

Example:

Intent: **find_flight**

"i would like to find a flight from charlotte to las vegas that makes a stop in st. louis"

○ ○ ○ ○ ○ ○ ○ ○ B-fromloc.city ○ B-toloc.city I-toloc.city ○ ○ ○ ○ ○ B-stoploc.city I-stoploc.city

The first experiments showed that addressing jointly improved both tasks performance

Joint Intent Detection and Slot filling

Now consider solving both tasks at once

We need a system that learns:

- The conditional probability of intent given the sentence representations

Joint Intent Detection and Slot filling

Now consider solving both tasks at once

We need a system that learns:

- The conditional probability of intent given the sentence representations
- The conditional probability of slot labels given the token representations, including across spans

Joint Intent Detection and Slot filling

Now consider solving both tasks at once

We need a system that learns:

- The conditional probability of intent given the sentence representations
- The conditional probability of slot labels given the token representations, including across spans
- The slot label dependency distributions

Joint Intent Detection and Slot filling

Now consider solving both tasks at once

We need a system that learns:

- The conditional probability of intent given the sentence representations
- The conditional probability of slot labels given the token representations, including across spans
- The label dependency distributions
- The joint distribution of intents and slot labels

Joint Intent Detection and Slot filling

Now consider solving both tasks at once

We need a system that learns:

- The conditional probability of intent given the sentence representations
- The conditional probability of slot labels given the token representations, including across spans
- The label dependency distributions
- The joint distribution of intents and slot labels

It is a difficult set of tasks, but:

- deep learning is here to help us, and
- the very first experiments showed that addressing the tasks jointly gave better results for each sub task than solving separately, or in series.

Deep learning - the very basics

A computer network that learns to map inputs to correct outputs

Prediction:

At each step the circuit tries to predict the output for the current input. It does this by applying a series of mathematical functions to the input in order. The functions contain numbers called weights and biases, collectively parameters.

The difference between the prediction and the correct output is encoded in something called the **loss function**.

After the prediction step we apply the derivative of the mathematical functions with respect to this loss function, in an analogue of the chain rule, to change the parameters within the network. This is called **back-propagation**.

We are thus changing the way we do things at each step based on the error we made at that step - we are learning!

Deep learning - the very basics - output


At the end of the circuit we make our predicted output by using a set of **classifiers**

A classifier produces a probability for each element of the set of possible classes - the function that does this is called **softmax**

To evaluate the difference between our prediction and the ground truth we use a **cross entropy** loss function

This measures a distance between our predicted probability vector across all classes, and a vector with 100% probability applied to the correct class and zero elsewhere.

$$L_{\text{cross-entropy}}(\hat{\mathbf{y}}, \mathbf{y}) = - \sum_i y_i \log(\hat{y}_i)$$

$\hat{\mathbf{y}}$	cross entropy	\mathbf{y}
0.1		0
0.03		0
0.02		0
0.7		1
0.01		0
0.05		0
0.09		0
		0

NLP - the very basics (inputs, feature engineering)

Only numbers can pass through computer networks

How do we represent words, and sentences?

Classically ensembles of words, sub-words and characters, syntactic (POS, dependency parsing)

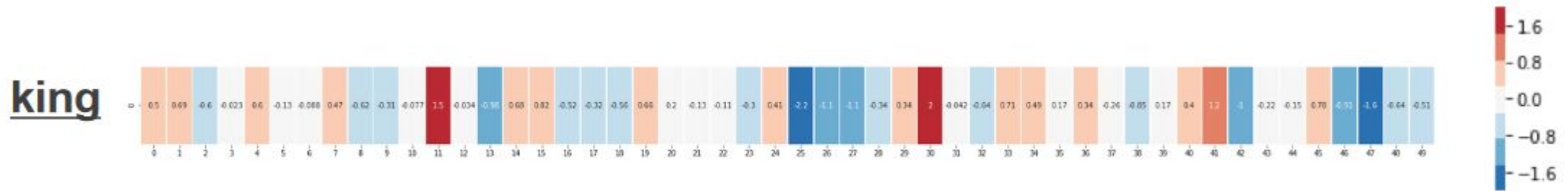
NLP - the very basics (inputs, feature engineering)

How do we represent words, and sentences?

Now we use vector representations of words, called embeddings as we embed the word in d dimensional vector space

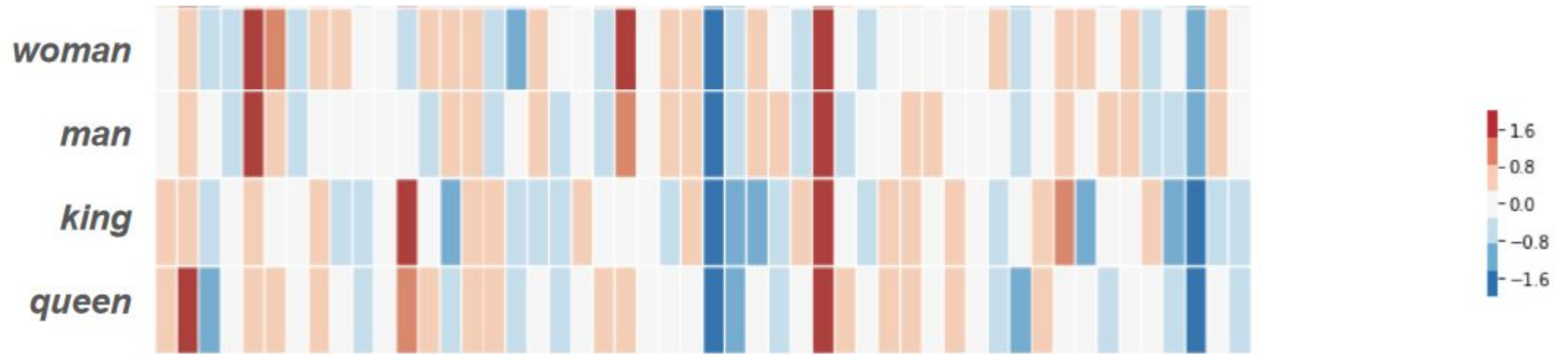
- One hot encoding - no context information
- Word2vec, fastText, Glove etc - some context information but usually only one representation per word
- BERT (Bidirectional Encoder Representations from Transformers) - different embeddings for different contexts of words

NLP - word embedding



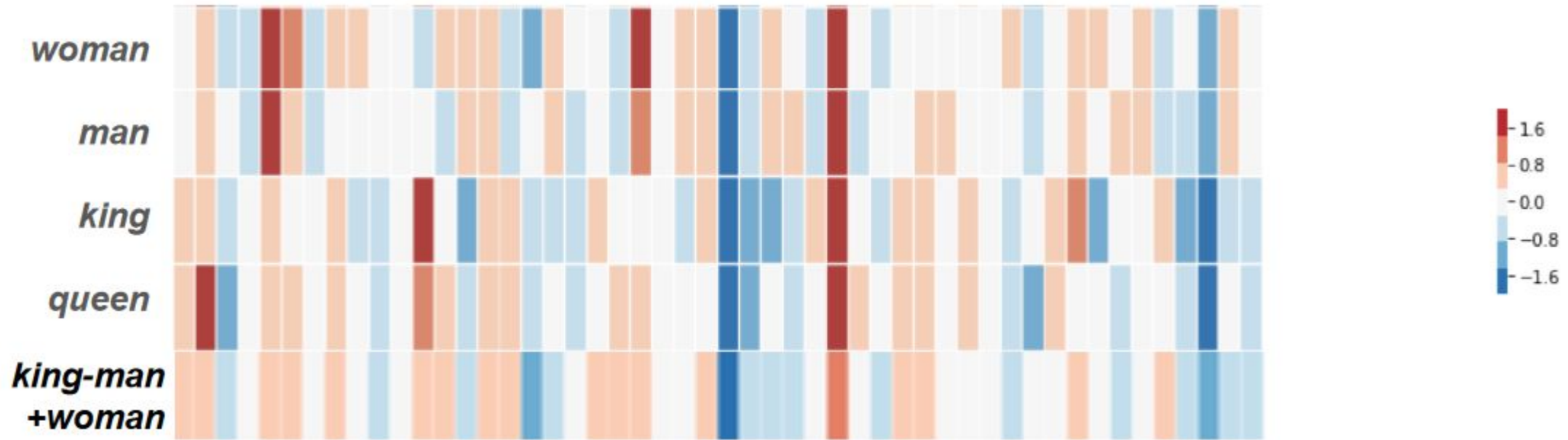
Dimension 50 GloVe embedding for the word **king**

NLP - word embedding

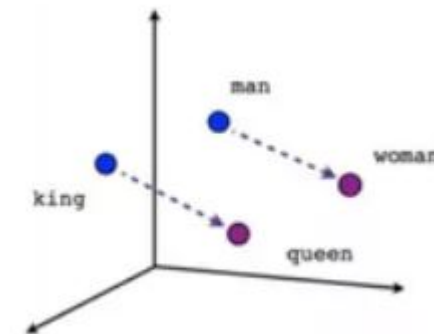


Similar words live in the same part of the vector space

NLP - word embedding



Word algebra: $\text{king} - \text{man} + \text{woman} \sim \text{queen}$

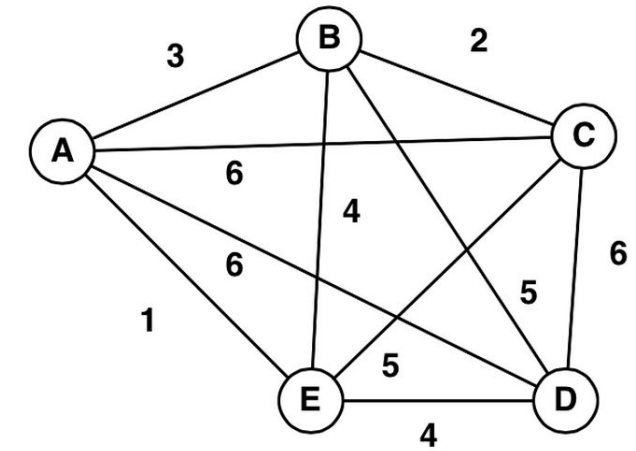


NLP - Other features

We can also embed sentences, paragraphs, entire documents as vectors.

What else might we want to use as inputs to this task?

NLP - Knowledge bases



What about the priors of slot label co-occurrence? Or intent and slot-label co-occurrence? Or word and slot-label co-occurrence? These kinds of things can be stored in **knowledge bases**.

We can encode such information in Graphs. A graph is a set of nodes and edges between them. The edges can be marked with a value.

e.g. slot label co-occurrence: the set of slot labels are the nodes, an edge between two labels is marked by the frequency (or relative frequency) that the two labels co-occur in sentences in the training **corpus***

* A **corpus** is a collection of documents

NLP - the very basics

We can also encode a graph into a set of vectors - e.g. Graph Convolutional Network (GCN).

Our learning network can apply **attention** between the training samples and the knowledge base.

An **architecture** describes a type of neural network which is used to solve a machine learning problem, and the way the component sub-architectures are arranged.

An abstract background featuring overlapping, semi-transparent teal-colored geometric shapes, primarily triangles and polygons, creating a layered, crystalline effect. The shapes are concentrated in the upper half of the slide, with some extending towards the right edge.

Joint SLU Approaches

14:00 - 14:30

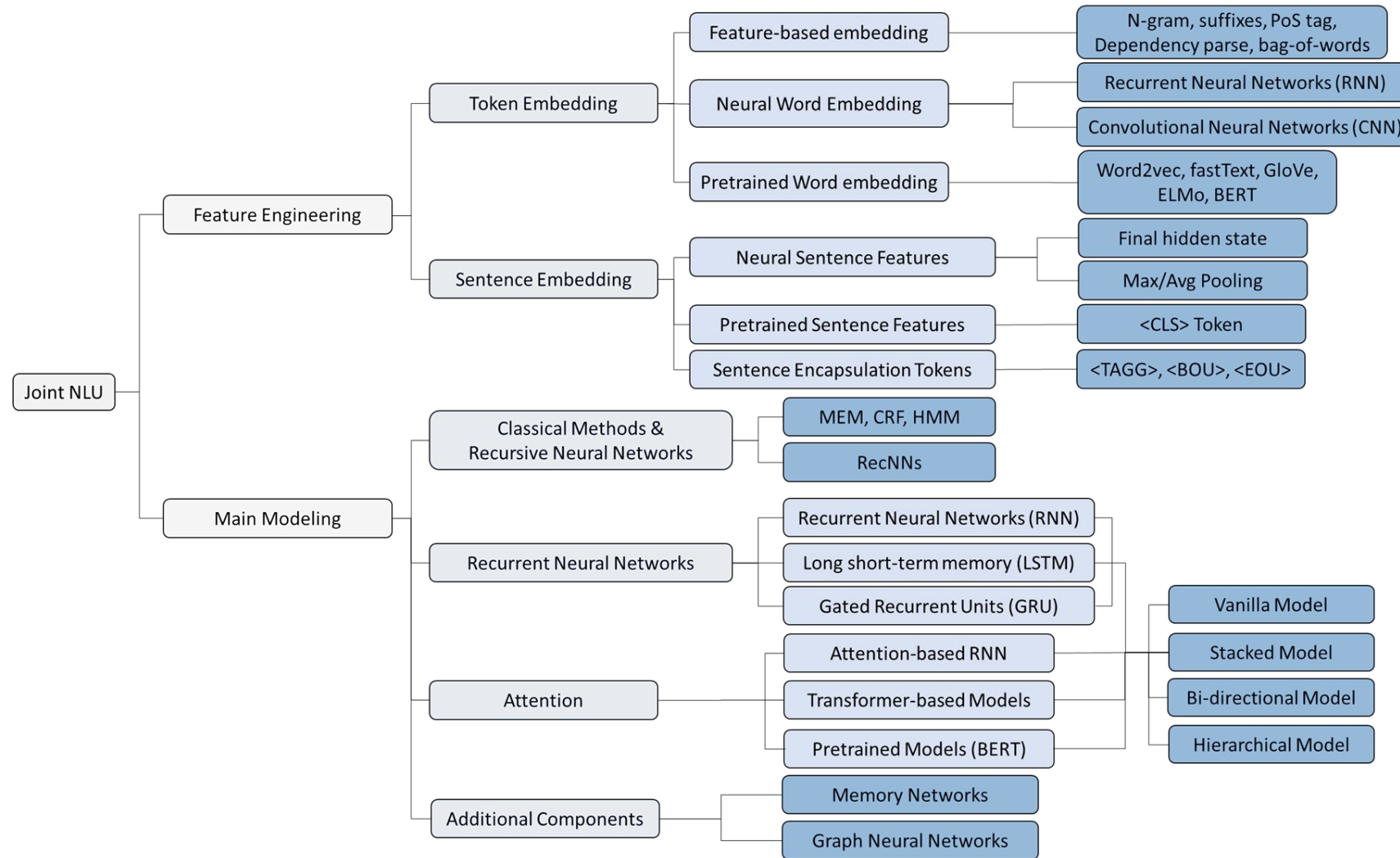
Joint intent detection and slot filling architecture overview

Table 4. Historical overview of joint task papers

Year	# papers	Feature engineering	Technologies
2008	1	words/n-grams/suffixes	CRF
2009	1	semantic tree	SVM
2013	1	CNN	CRF
2014	1	dependency parse	RecNN (diff to RNN)
2015	1	RNN words, CNN sentence, bag-of-words	MLP
2016	6	RNN, K-SAN	(Bi)LSTM, (Bi)GRU, encoder-decoder RNN, attention
2017	4	character, word, CNN	BiLSTM
2018	18	word2vec, GloVe, ELMo, CNN sentence, attention sentence	BiLSTM, BiGRU, encoder-decoder RNN, Capsule NN, BiDirectional
2019	29	BERT, GloVe, character, knowledge base (tuples), delexicalisation	memory NN, transformer, CRF, attention, BiDirectional
2020	10	BERT, Graph embedding	Graph S-LSTM, BiDirectional, GCN, Capsule

Weld et al, 2022

Technological approaches in NLU



Weld et al, 2022

Hidden Markov Model (HMM)

Conditional Random Field (CRF) - label dependency

HMM Assumption: a hidden sequence (the slot labels) is driving an observable sequence (the words)

It uses prior distributions of label sequences and conditional probabilities of words given labels to construct the most likely sequence of labels to have given us the sequence of words we observe.

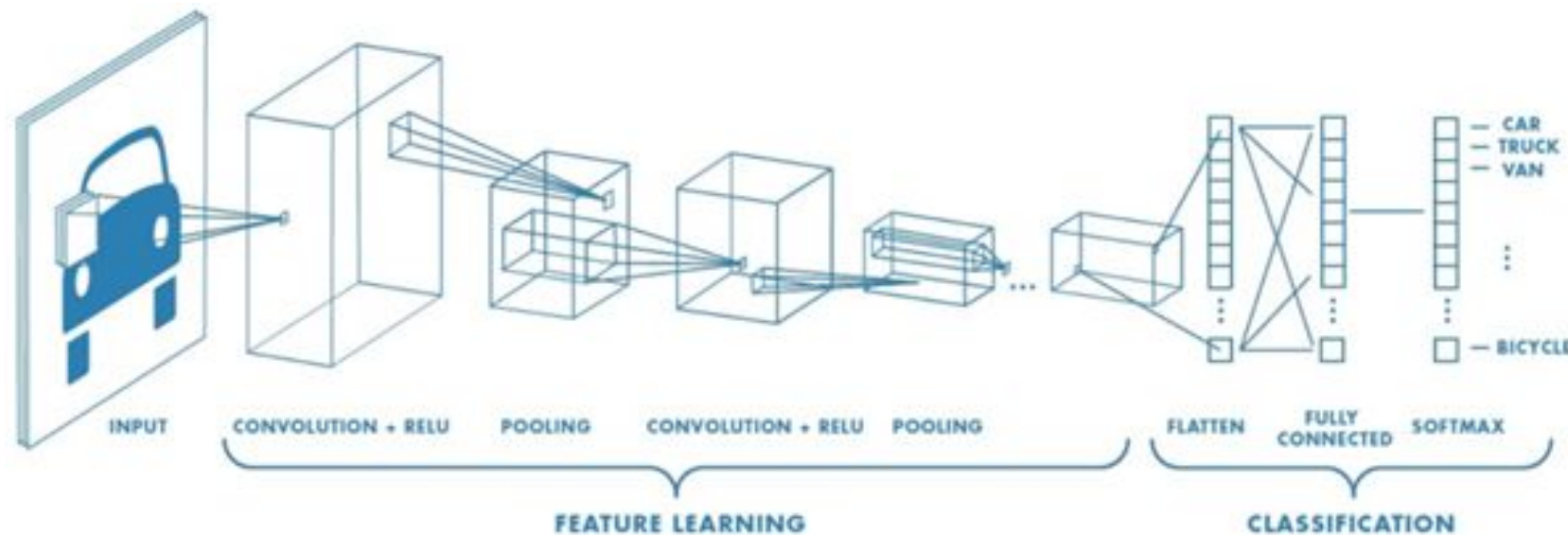
CRF is similar and learns these conditional probabilities to produce most likely sequences of labels in a decoding step

It is a classical statistical methodology but is effective and even today deep learning models will attach a CRF at the end of their models to improve performance on label dependency.

Convolutional Neural Network - CNN

From image analysis and classification

Features are created by passing filters over the image and performing linear and non-linear operations on them.



Recurrent Neural Network (RNN)

Intuitively how do you analyse language as it arrives at your ear?

Is sequence important or just the presence of words?

Do you scan the words in both directions in your mind to find meaning?

Recurrent Neural Network (RNN)

Considers the sequence that information arrives in analysis

A hidden state is updated at each time step with its current value and the newly arrived information (word) as inputs.

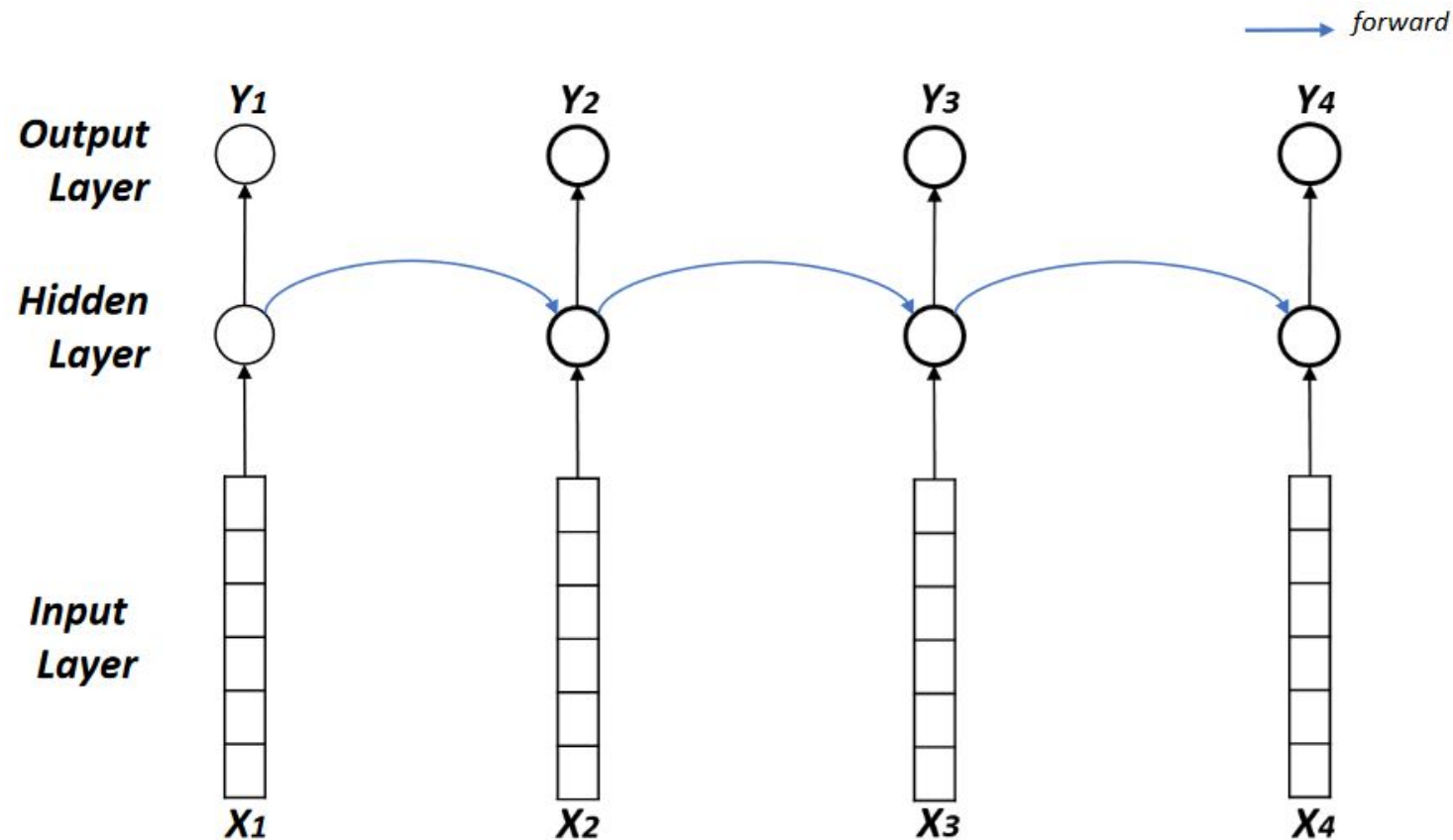
At each step the hidden state has encapsulated the words that have arrived up to that point. We can use these intermediate states for token (slot) labelling

The final hidden state has encapsulated the full sentence that has arrived with regard to the order the words arrived. We can use this for sentence (intent) classification

Bidirectional RNN considers the word sequence and the reverse word sequence.

Recurrent Neural Network (RNN)

Considers the sequence that information arrives in analysis



LSTM and GRU

Two specially designed RNNs that address issues with:

- long term dependency - the loss of important effect between two sub-sentences that are far away from each other
- vanishing gradient - the hidden state only changes minimally as the sentence progresses

Hierarchical models - Capsule networks

Consist of layers of capsules which processes information as it comes in at the lowest layer once a signal of sufficient strength is built a capsule can pass information to the next layer.

Memory networks

Similar to capsule but not as hierarchical, the slot memory blocks and intent memory blocks diffuse information between themselves as new words are encountered in sequence.

Attention Networks

We represent subsections of an input by vectors of numbers e.g. word embeddings of tokens, knowledge base embedding

We then process these into other representational vectors e.g. by CNN or RNN

We can perform vector operations between each pair of vectors e.g. dot product

Our network can learn which pairs are of differing relative importance to our task.

This is **attention**.

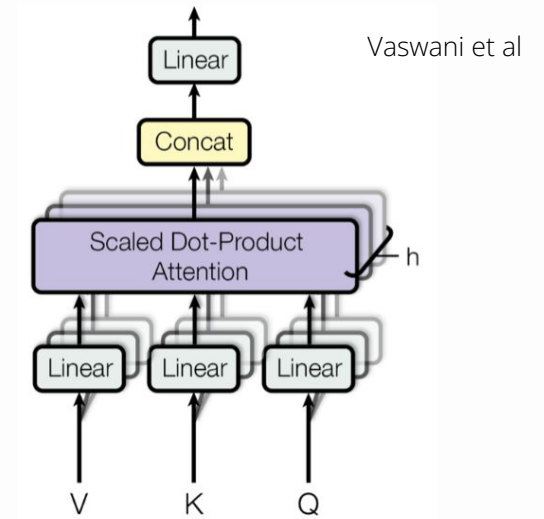
The attention output can be combined with the word representations to inform our task, or....

Transformer

“Attention Is All You Need”

Multi-head attention only networks

No recurrence, it just takes the whole input and performs all the pairwise attentions.

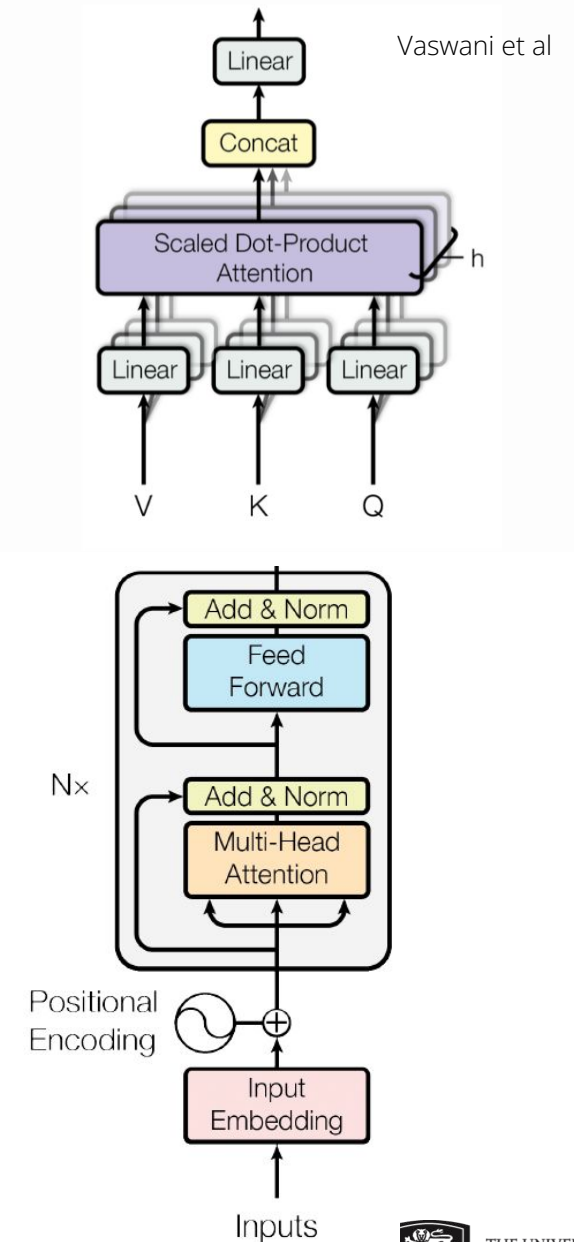


Transformer

It also does this over multi-heads, in effect each head learning to focus on different aspects of the input.

Intuitively when you understand a sentence you are concentrating on the syntax, the semantics, weighing the importance of the clauses, looking for irony or sarcasm, perhaps reading body language, etc.

Recurrence is removed, each word token is also given a positional encoding so some information on where it fits in the sequence is maintained.



BERT (and GPT3) and other pre-trained models

BERT is a word embedding technology that uses a Transformer encoder.

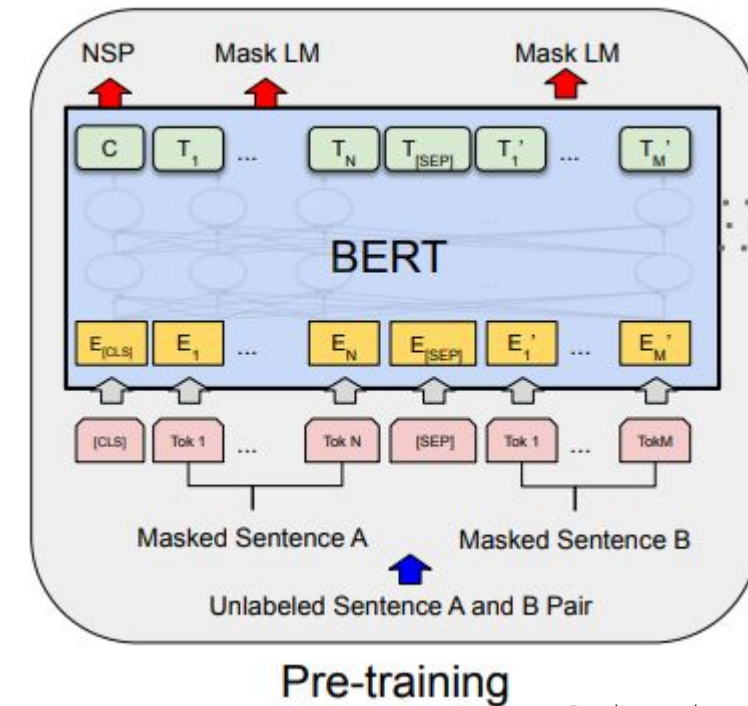
[CLS] Find me an Indian restaurant [SEP] Make a booking for Tuesday night [SEP]

It is trained on pairs of sentences on two tasks:

- Masked Language Modelling - hide about 20% of the words and learn to predict them
- Next Sentence Prediction - does the second sentence follow the first sentence in the training corpus

By training on these tasks the model learns about the language* - words that tend to appear together, different contexts that a word may appear in and something about longer inter-sentence interactions.

*the language of the training corpus - BERT uses Wikipedia, GPT3 uses a larger corpus of online text data and carries language bias from the corpus



Devlin et al

BERT (and GPT3) and other pre-trained models

After pre-training I can pass sentences through my pre-trained model and it will give me embeddings for each word in the sentence.

If I pass in the sentences:

[CLS] please ring the office [SEP]

[CLS] i wear a gold ring [SEP]

I will get different embeddings for the word ring because the model incorporates context.

Note the [CLS] token - it is there to get an embedding for the whole sentence (CLS stands for classifier).

BERT and his cousins

BERT has revolutionised many tasks within NLP

It spawned many progressions - ROBERTA, Alberta, Longformer

These make BERT more efficient, or train on different tasks, or on longer inputs, or different languages (CamemBERT for French), or on multiple languages (mBERT).

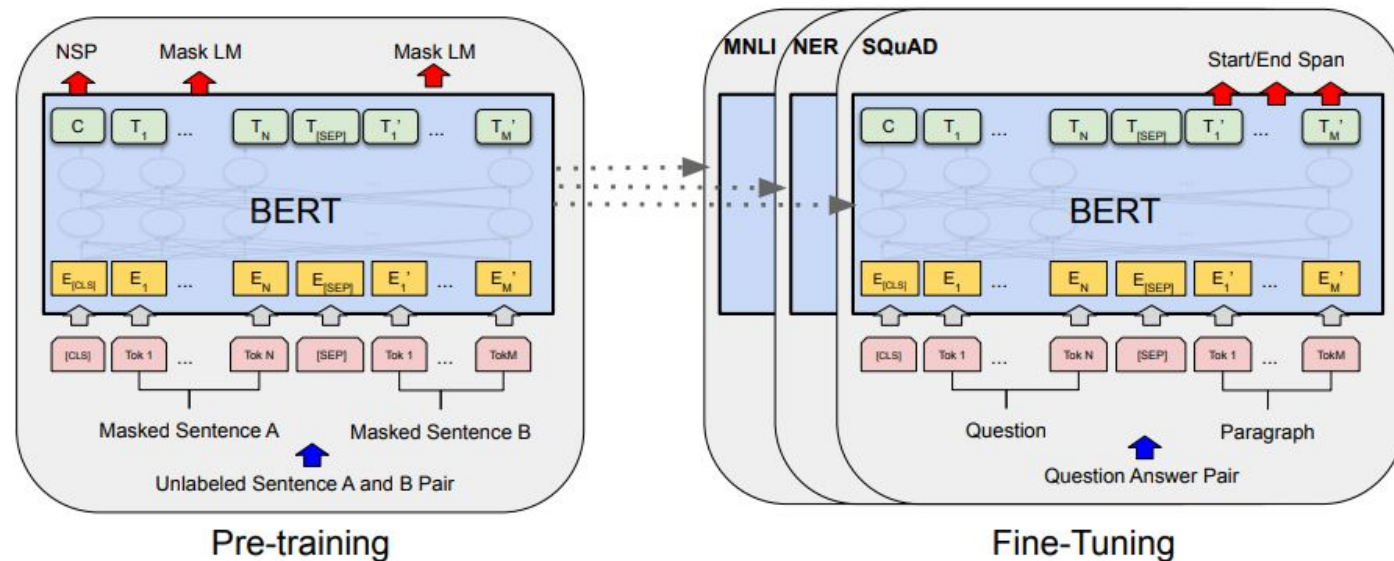
Using pre-trained models

We use the representations produced by pre-trained models as features in downstream tasks

We can freeze the representations coming from the pre-trained model

or

We can allow the pre-trained model to change its representations by back propagating from the downstream task



Devlin et al

Now let's look at some joint ID and SF networks

The following are earlier state-of-the-art joint intent classification and slot filling models

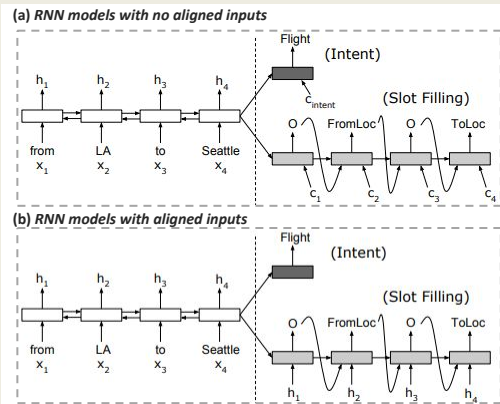
- They discovered that an utterance-level intent and word-level slot labels have high correlation with each other

e.g. AddToPlaylist

e.g. playlist, music-item

Attention RNNs

(Liu and Lane, 2016)

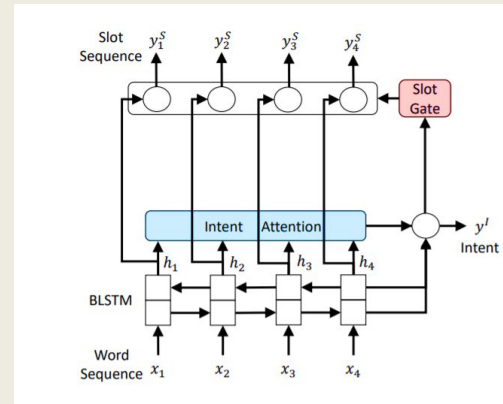


Attention-based encoder-decoder

trained in a single attention-based RNN model with a shared objective.

Slot-gated Model

(Goo et al., 2018)

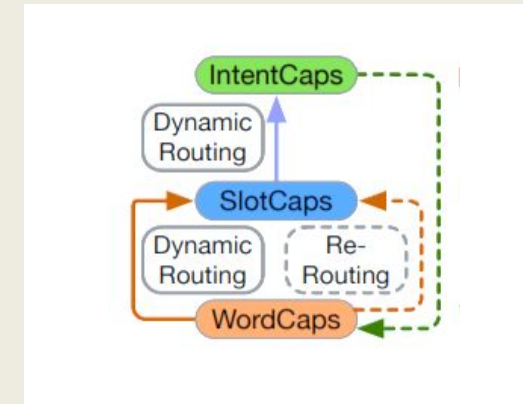


Slot-gated Intent Attention

Use BiLSTM to integrate the detected intent vector to the slot filling task

Capsule Neural Networks

(Zhang et al., 2019)



Slot-gated Intent Attention

Use capsules, achieve slot filling and intent detection via a dynamic rerouting-by-agreement schema

Now let's look at some joint ID and SF networks

The following are earlier state-of-the-art joint intent classification and slot filling models

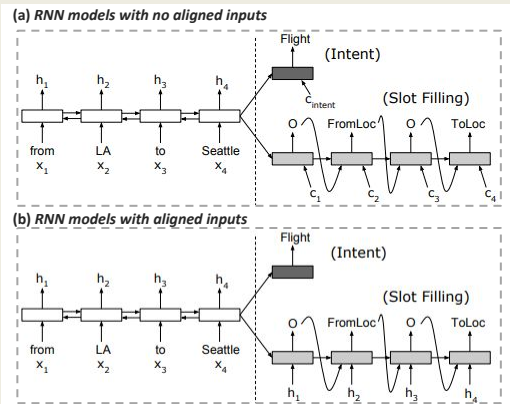
- They discovered that an utterance-level intent and word-level slot labels have high correlation with each other

e.g. AddToPlaylist

e.g. playlist, music-item

Attention RNNs

(Liu and Lane, 2016)



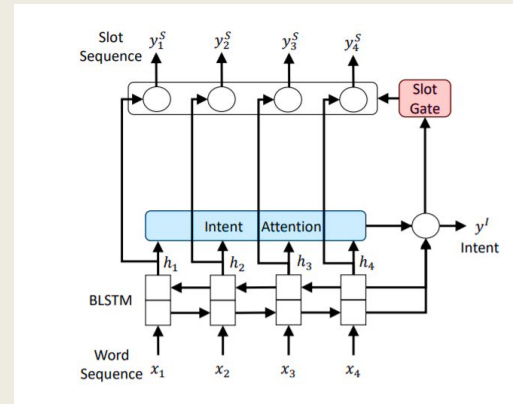
Attention-based encoder-decoder

trained in a single attention-based RNN model with a shared objective.

Implicit Join

Slot-gated Model

(Goo et al., 2018)



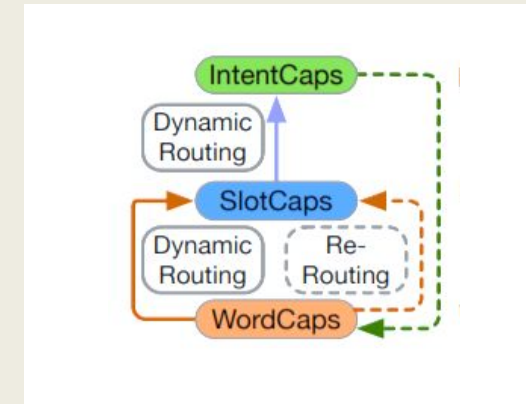
Slot-gated Intent Attention

Use BiLSTM to integrate the detected intent vector to the slot filling task

Intent to slot only

Capsule Neural Networks

(Zhang et al., 2019)

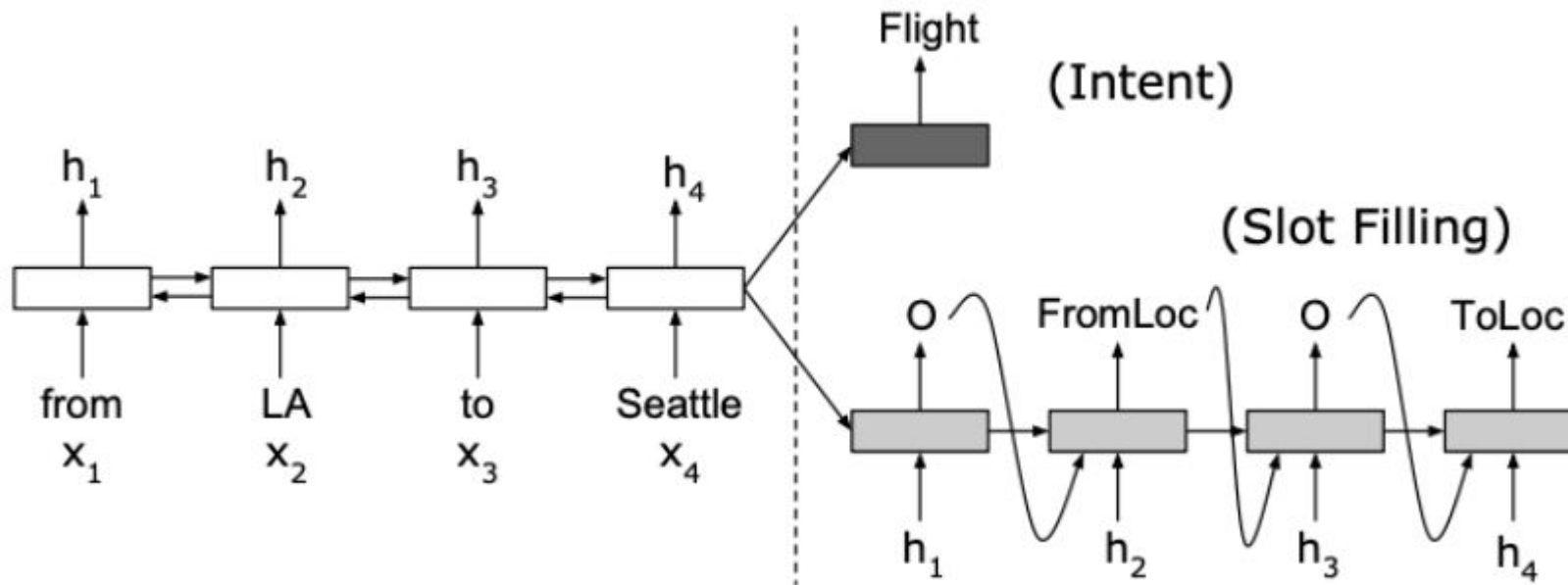


Slot-gated Intent Attention

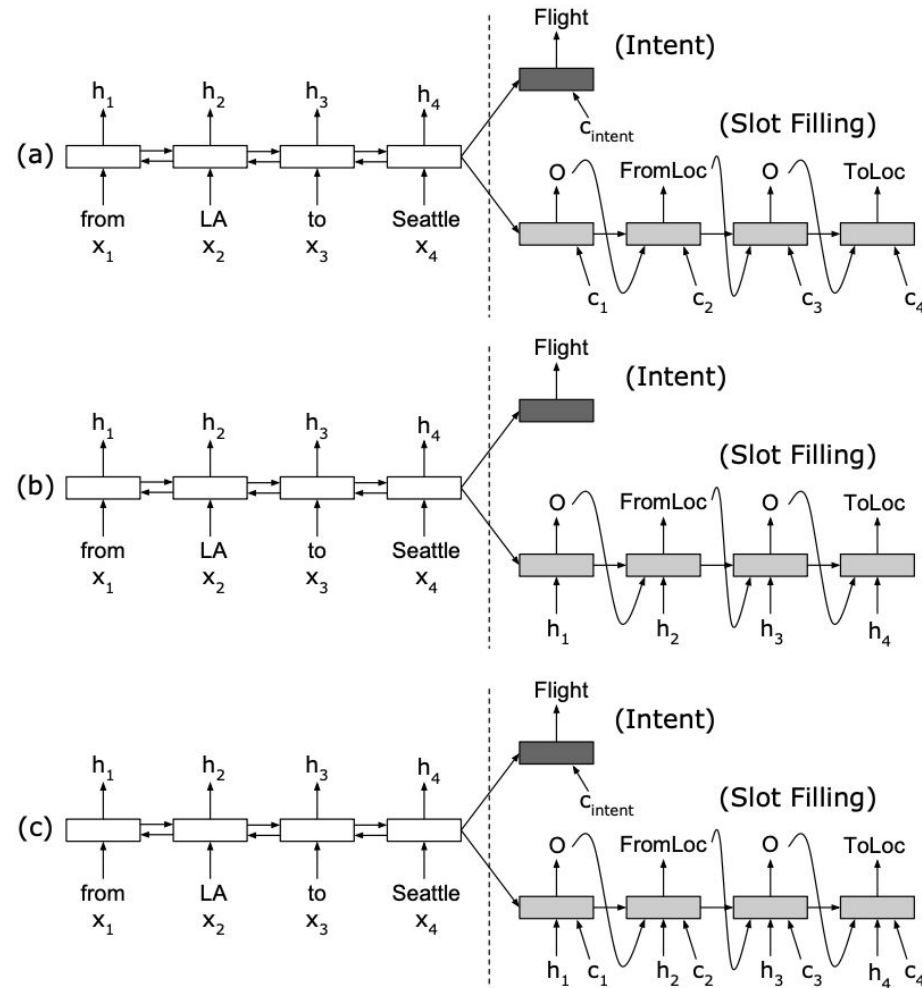
Use capsules, achieve slot filling and intent detection via a dynamic rerouting-by-agreement schema

Non-direct Join

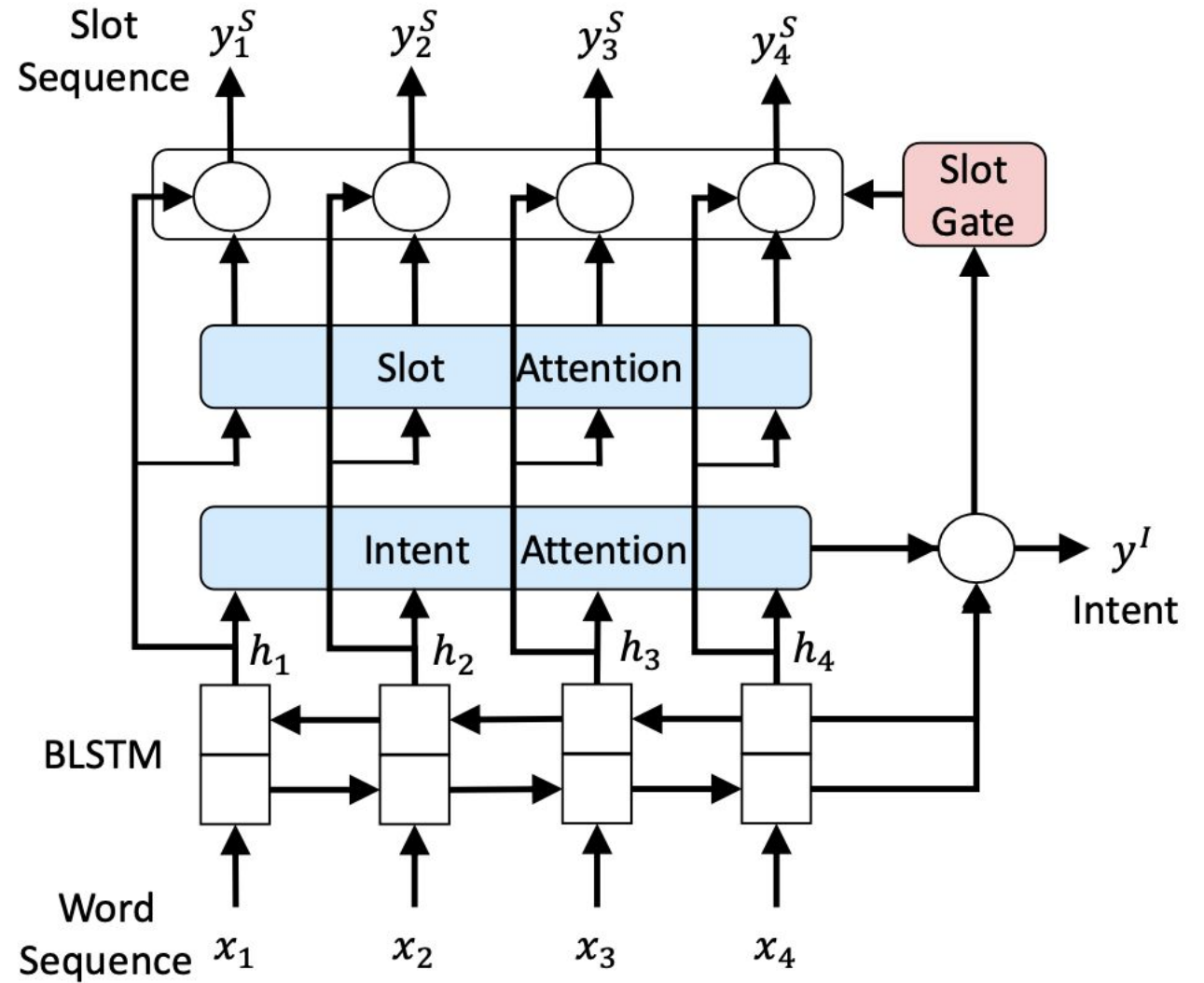
Attention RNNs (Liu and Lane 2016)



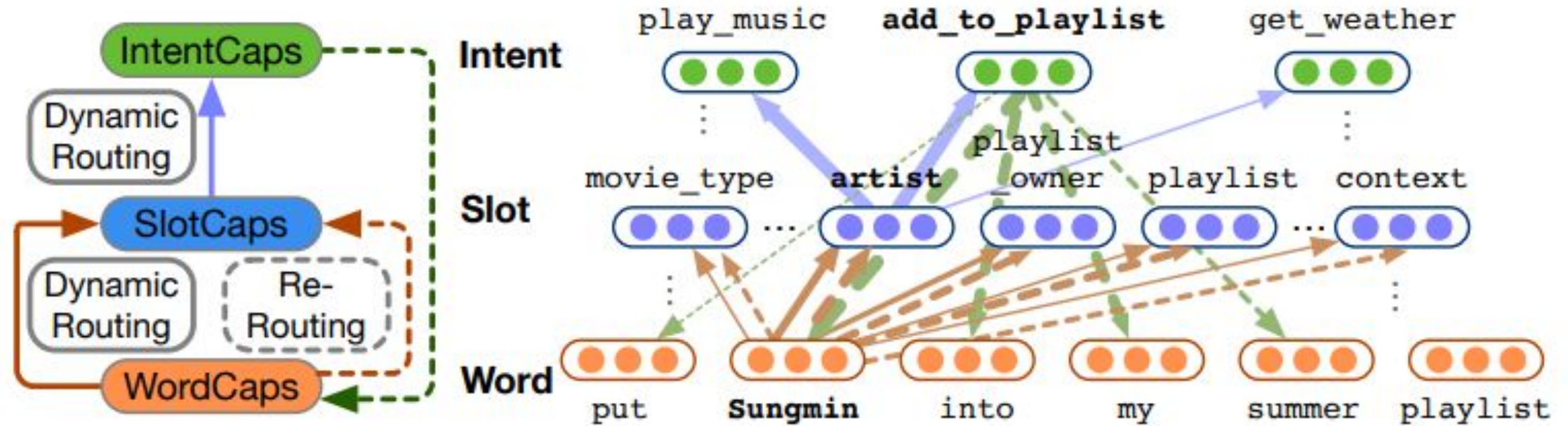
Attention RNNs (Liu and Lane 2016)



Slot gate (Goo et al, 2018)



Capsule network (Zhang et al, 2019)



Joint BERT (Chen et al, 2019)

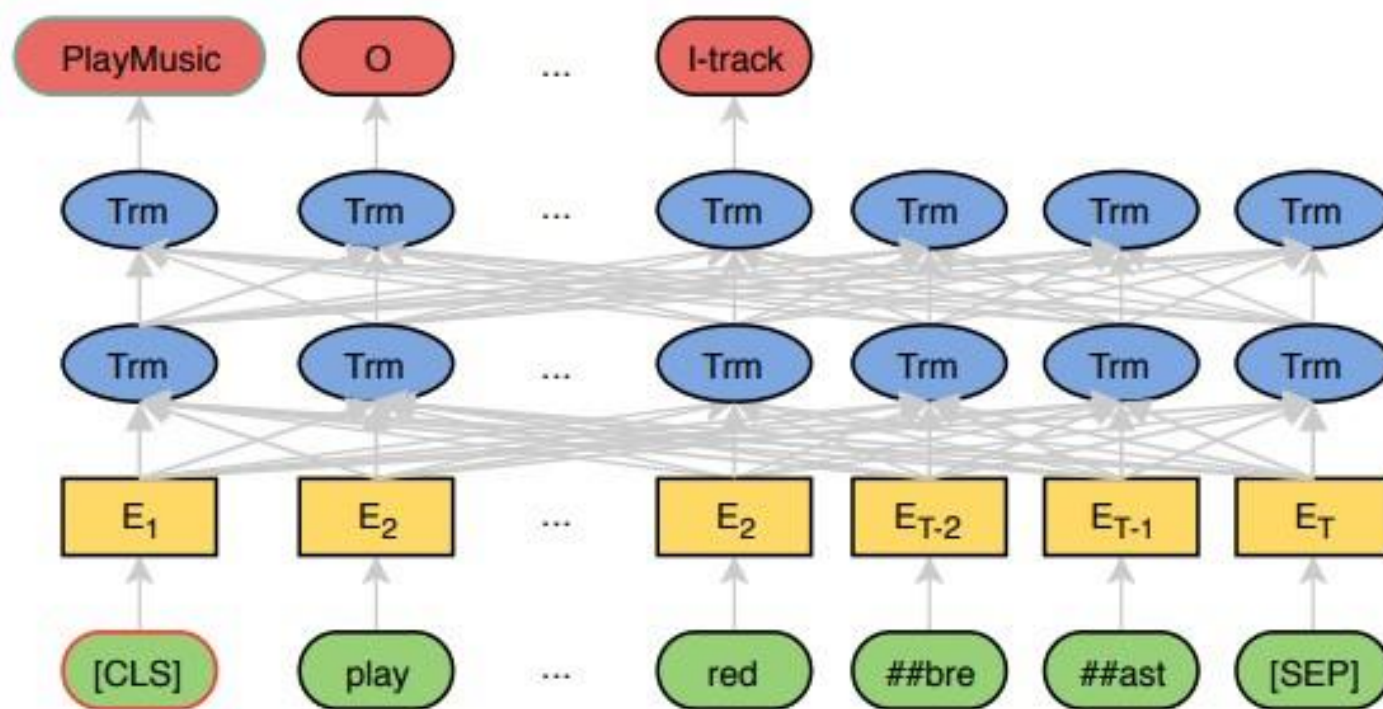
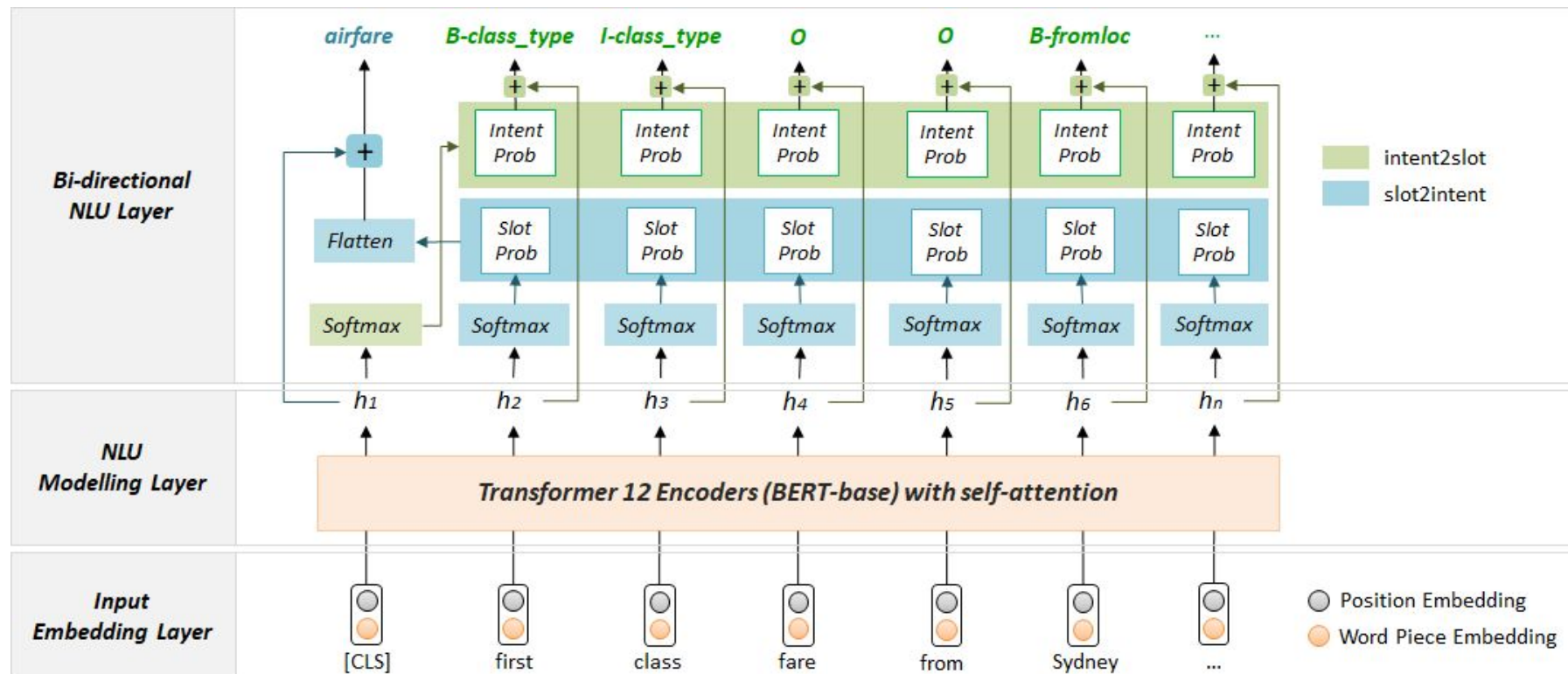


Figure 1: A high-level view of the proposed model. The input query is “play the song little robin redbreast”.

Bidirectional models - Our Methodology: Bi-ED 2021

- Our Bi-directional 'Explicit and Direct' joint flow mechanism



Other approaches: Knowledge bases

Constructs containing information or statistical priors that may be useful to the task at hand.

They may be constructed independent of the task, or as a preliminary step using information from the training data.

They have been used for feature construction, as features themselves, and to be consulted via attention.

e.g. A graph reflecting interaction between words, slots, intents either in the general language or in the training set

Other approaches: Multi-task learning

Synergies with related tasks:

- Predicting user information from metadata (utterance level)
- Part-of-speech (POS) or Named Entity (NER) tagging at the token level
- Dialogue action prediction (downstream task)

Results show a parsimonious approach gives better results

The joint task is a multi-task approach

An abstract background featuring overlapping, translucent teal-colored geometric shapes, primarily triangles and polygons, creating a complex, crystalline pattern in the upper right corner of the slide.

Q'n'A and Break

14:30 - 14:45

An abstract background featuring overlapping, semi-transparent teal polygons of various shapes and sizes, creating a layered, crystalline effect. The polygons are primarily located in the upper half of the slide, with some extending towards the right edge.

Hands-on Exercise (Joint BERT)

14:45 - 15:00

An abstract background featuring overlapping, semi-transparent teal-colored geometric shapes, primarily triangles and polygons, creating a complex, crystalline structure. The shapes are layered, with some appearing more prominent than others, and they extend from the top and right edges of the frame towards the center.

SLU Evaluation

15:00 - 15:15

NLU datasets

Name	Public	Train-Val-Test	Num Intent	Num Slots	Domain, Notes
ATIS	Y	4478/500/893	21	128	air travel
SNIPS-NLU	Y	13084/700/700	7	72	personal assist.
FRAMES	Y	20006/-/6598	24	136	hotel, multiturn
CQUD	N	3286	43	20	Chinese, question answering
TREC	Y	5500/-/500	6(50)	-	question classification
TRAINS	N	5355/-/1336	12	32	problem solving, multiturn
Microsoft Cortana	N	10k/1k/15k	10-20	15-63	personal assist., multidomain
Facebook	Y	30521/4181/8621	12	11	multi-lingual task oriented
SRTS FrameNet	N	2803/-/312	12	61	robotics
Alexa	N	264000/-/-	246	3409	17 domains
DSTC2	Y	4790/1579/4485	13	9	multiturn, restaurant search
DSTC4	Y	5648/1939/3178	87	68	multiturn, tourism dialogue
DSTC5	Y	27528/3441/3447	84	533	dialogue with social robots
CMRS	N	2901/969/967	5	11	Chinese, room reservations
CU-Move	N	57584/-/-	5	38	in-vehicle dialogue
AMIE	N	3418/-/-	10	7	in-vehicle dialogue
TeleBank	N	2238/-/-	25	17	Korean, banking
CONDA	Y	26921/8974/8974	4	6	in-game chat
MTOP	Y	73174/10453/20907	117	78	11 domains, 6 languages
MIT Movie_Eng	Y	8798/97/2443	-	25	movies, slot only
MIT Restaurant	Y	6894/766/1521	-	17	restaurants, slot only



Evaluation metrics for slot filling - Span based f1, recall, precision

Add *Par For The Course* by *Aimee Mann* to my *Sad Songs* playlist

O B-song I-song I song-I-song O B-artist I-artist O O B-playlist I-playlist O

For a meta-class \$C\$ we define at the span level:

- TP is the number of spans of meta-class \$C\$ which are wholly correctly predicted;
- FP is the number of spans of a different meta-class which are incorrectly predicted as being of meta-class \$C\$;
- FN is the number of spans of meta-class \$C\$ which are incorrectly predicted, partially or wholly, to another meta-class.

Precision = $TP / (TP + FP)$

Recall = $TP / (TP + FN)$

f1 = $2PR / (P+R)$

Report macro averaging on non-O slots

Evaluation metrics

Intent classification - accuracy

Joint measure - sentence accuracy, a sentence is correctly annotated if intent and all slot tags are correct

Performance Evaluation

<i>Model</i>	<i>ATIS (10 epoch)</i>			<i>SNIPS (20 epoch)</i>		
	<i>Slot (F1)</i>	<i>Intent (acc)</i>	<i>Sentence (acc)</i>	<i>Slot (F1)</i>	<i>Intent (acc)</i>	<i>Sentence (acc)</i>
RNN-LSTM (Hakkani-Tür et al., 2016)	94.3	92.6	80.7	87.3	96.9	73.4
Attention Bi-RNN (Liu and Lane, 2016)	94.2	91.1	78.9	87.8	96.7	74.1
Slot-gated Intent Attention (Goo et al., 2018)	95.2	94.1	82.6	88.3	96.8	74.6
Slot-gated Full Attention (Goo et al., 2018)	94.8	93.6	82.2	88.8	97.0	75.5
Capsule NLU (Zhang et al., 2019)	95.2	95.0	83.4	91.8	97.3	80.9
Bidirectional LSTM-CRF (Haihong et al., 2019)	95.8	97.8	86.8	91.4	97.4	80.6
Joint BERT (Chen et al., 2020)	96.1	97.5	88.2	97.0	98.6	92.8
Stack-Prop. (Qin et al., 2019)	96.1	97.5	88.6	97.0	99.2	92.9
Our Model with Slot2Intent Only	95.5	97.8	87.5	95.4	98.3	89.4
Our Model with Intent2Slot Only	95.7	97.8	87.5	95.0	98.1	88.6
Our Model with Both Bi-directional Flow	96.3	98.6	88.6	97.2	99.2	92.8

Datasets

ATIS and SNIPS are a solid benchmark and provide some variation

- single vs multi domain
- balanced vs unbalanced

Problems

- Limited scope
- Limited sentence structures, unnatural language
- Fully annotated - cost of annotation, accuracy of annotation

Other datasets - introducing CONDA

Toxicity

- behaviour intended to insult or humiliate
- problematic to the gaming industry
- problematic to online discourse

Identification is the first step

Natural language understanding

- intent detection
- hierarchical models (joint intent and slot detection)
- rich literature of models and methods

To provide a hierarchically annotated in-game contextual dataset for identifying and understanding toxicity, with distinctions from and similarities to existing datasets

Language warning

DOTA 2

- Multiplayer online game
- 2 teams of 5 players choose characters and characteristics
- Attempt to win by destroying the other team's 'ancient' structure
- Chat functionality



Sample chat 1

roam mirana?	-68
ye mirana [SEPA] will u roam?	-54
No	-49
nono	-48
mirana core	-46
stfu brood nobody speaks with last pick brood	-30
;## [SEPA] i will fuck you	-25
no u dont [SEPA] u will suffer and your teammates will blame u	-19
^^ [SEPA] but [SEPA] First time brood [SEPA] Dont care	-3
oh easy report then [SEPA] what was the point of that	16
gg	72
Wtf	162
well thats a RQ	166
your mom	166
you arent [SEPA] even funny [SEPA] stop trying so hard	166
Rq shits	179
[SEPA] why man	221
XDD	266
told u	450
DIE SHIT	463
lvl 1 starfall [SEPA] what a joke	891
lol	1026
FTW	1797
this team [SEPA] is pathetic	2157
can you give me a courier for challenge? :P	2180
not yet [SEPA] but soon	2193
RUBICK [SEPA] I SAY U [SEPA] U LOST	2201
please brood shhh [SEPA] u are shit	2211
;3 [SEPA] <3	2238

Pre-game: discuss tactics, already in-fighting

During game: short

Post game: team lost, recrimination

Tokenisation and annotation

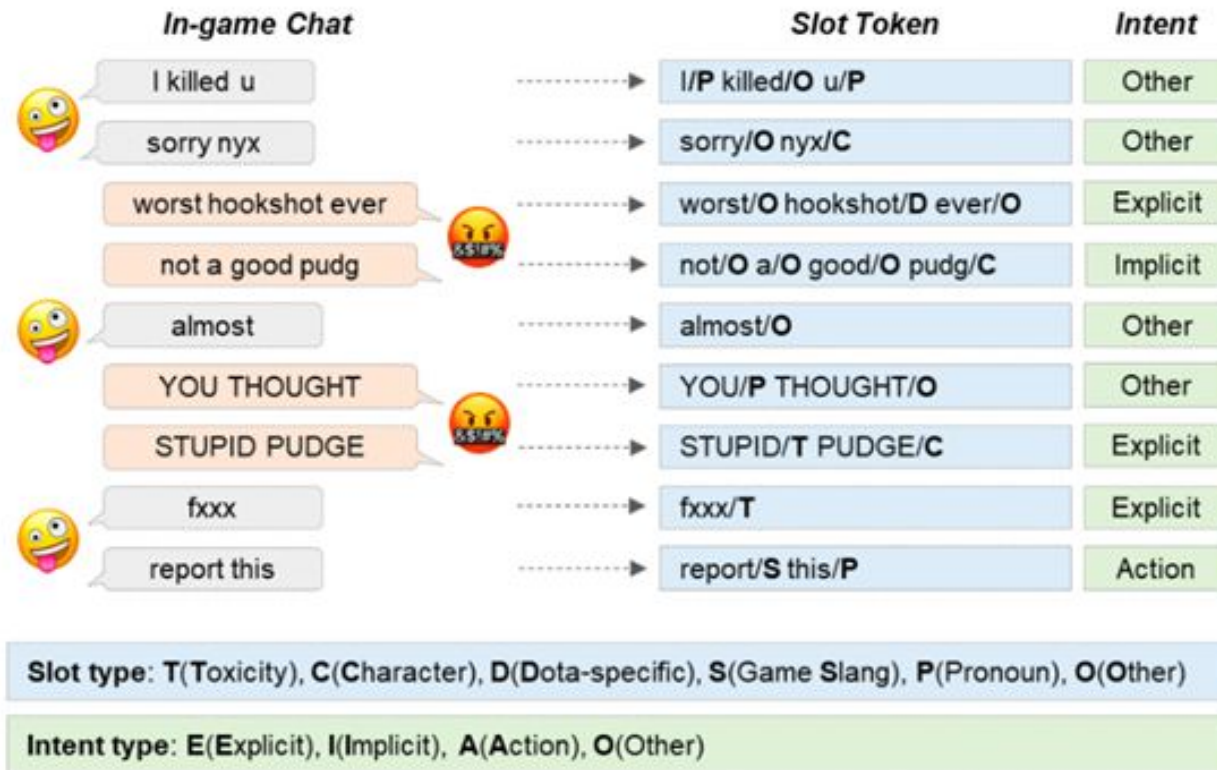



Figure 1: An example intent/slot annotation from the CONDA (CONtextual Dual-Annotated) dataset.

Experiments and metrics

Model	Metrics												
	UCA	U-F1(E)	U-F1(I)	U-F1(A)	U-F1(O)	T-F1	T-F1(T)	T-F1(S)	T-F1(C)	T-F1(D)	T-F1(P)	T-F1(O)	JSA
RNN-NLU (Liu and Lane, 2016)	0.905	0.813	0.720	0.783	0.944	0.970	0.931	0.981	0.930	0.718	0.991	0.987	0.854
Slot-gated (Goo et al., 2018)	0.894	0.806	0.694	0.773	0.938	0.991	0.978	0.992	0.982	0.952	0.997	0.994	0.875
Inter-BiLSTM (Wang et al., 2018)	0.869	0.719	0.590	0.728	0.923	0.865	0.871	0.889	0.869	0.788	0.942	0.924	0.711
Capsule NN (Zhang et al., 2019a)	0.876	0.735	0.706	0.643	0.926	0.991	0.975	0.991	0.982	0.949	0.997	0.994	0.855
Joint BERT (Castellucci et al., 2019)	0.921	0.872	0.768	0.800	0.954	0.989	0.972	0.992	0.979	0.914	0.998	0.993	0.895

Table 5: Joint intent classification and slot labeling performance on CONDA for the five NLU baseline models. It is measured in the four multi-level metrics including: UCA (Utterance Classification Accuracy); the break-down U-F1 for each intent class - E (Explicit), I (Implicit), A (Action), O (Other); the overall T-F1 and breakdown for each slot class - T (Toxicity), S (game Slang), C (Character), D (Dota-specific), P (Pronoun), O (Other); and JSA (Joint Semantic Accuracy).

An abstract background composed of overlapping, semi-transparent teal-colored triangles and polygons, creating a complex, low-poly geometric pattern that resembles a stylized mountain range or a network structure.

Hands-on Exercise (Datasets, Metrics, Experiments)

15:15 - 16:00

An abstract graphic in the top right corner of the slide. It consists of several overlapping, semi-transparent teal-colored polygons, primarily triangles and quadrilaterals, of various sizes. These shapes are arranged to create a sense of depth and geometric complexity, resembling a low-poly landscape or a cluster of crystals. The colors range from a light, pale teal to a darker, more saturated teal.

Future Directions

16:00 - 16:15

Open issues in NLU

The standard experiment on the standard datasets is “solved”

Multi-turn datasets

Multi-intent utterances

Evolving intents and slots

Generalisability - New domains

Generalisability - New languages

Limited annotated training data - Zero and few shot learning

Open issues - the standard dataset are “solved”

Table 7. NLU performance on ATIS and SNIPS-NLU data sets (%). * denotes ATIS 10 epoch, SNIPS 20 epoch, i denotes epoch count implied, † indicates GitHub available, – denotes not reported

Paper, Model	ATIS			SNIPS		
	Slot f1	Int acc	Sem acc	Slot f1	Int acc	Sem acc
[98] (2020) SASGBC	96.69	98.21	91.6	96.43	98.86	92.57
[89] (2020) fully-E@EMG-CRF	96.4	99.0	89.6	97.2	99.7	93.6
[33] (2021)	96.3	98.6	88.6	97.2	99.2	92.8
[38] (2021)	96.4	98.2	88.5	97.6	99.3	93.0
[70] (2021) BERT	97.1	98.8	93.1	96.1	98.0	88.8
[90] (2021)	97.3	98.3	90.2	98.3	98.9	90.2

Single domain, single utterance, task focused - useful, but limited in scope

New datasets, new domains

Multi-turn, multi-language, code switching (language changing), multi-modal

Conversational, instructional, third party (monitoring conversation)

Open issues - Multi-turn datasets

Multi-turn datasets (remember our eating and shopping in Singapore example)

Intuitively, how do you keep track of a conversation?

- Feed recent history into predictions for the current utterance
- Recent history can include sentence embeddings, domain and intent predictions, slot labels encountered.
- The history can be a feature or used for attention with the current utterance

Open issues - Multi-intent learning

“Find me a flight to Auckland tomorrow and a hotel near the airport”

Early approaches just merged multi-intents into one new intent: find_flight and find_hotel became find_flight_hotel

Multi-label classification is a well understood problem in machine learning with top-K algorithms for example being used

MixATIS is a multi-intent version of ATIS

Open issues - Generalisation

Deployed models show a drop off from experimental performance

New intents and slots appear

Language changes

Transfer to a new domain/language with no annotated training data (annotation is expensive)

Some solutions:

- ensemble models
- delexicalisation (replacing words with generic tokens in training)
- pre-trained models
- translation of training datasets (multi-ATIS exists)
- architecture and weight transfer
- few and zero-shot learning
- meta-learning

Other NLP applications

Multi-word expressions

This field recently introduced a sentence level classification task to their token level task of identifying MWEs (e.g. idioms)

Sentence	Id- iomatic
When removing a <u>big fish</u> from a net, it should be held in a <u>manner</u> that supports the girth. (newsdakota.com)	No
It was still a respectable finish for both Fadol and Nayre, who were ranked outside the top 500 in the world but caught some <u>big fish</u> along the way. (philstar.com)	Yes
To pay attention only to new housing and houses I think skews the <u>big picture</u> . (streets.mn)	Yes

The future of NLU

Multi-modal

- image, video, other sensor data


Explainable NLU

- attention maps
- separation of the encoding for each task has started to take place
- experiments varying one of these encodings then measuring the effect on single or joint metrics

Extension of dual level annotation and joint tasks to new fields

References

1. Tomas Mikolov, Kai Chen, Gregory S. Corrado, and Jeffrey Dean. 2013. Efficient estimation of word representations in vector space. *CoRR*, abs/1301.3781.
2. Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. Bert: Pre-training of deep bidirectional transformers for language understanding. In Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers), pages 4171–4186.
3. Bing Liu and Ian Lane. 2016. Attention-Based Recurrent Neural Network Models for Joint Intent Detection and Slot Filling. In Interspeech 2016. ISCA, San Francisco, USA, 685–689. <https://doi.org/10.21437/Interspeech.2016-1352>
4. Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In Advances in neural information processing systems, pages 5998–6008
5. Chih-Wen Goo, Guang Gao, Yun-Kai Hsu, Chih-Li Huo, Tsung-Chieh Chen, Keng-Wei Hsu, and Yun-Nung Chen. 2018. Slot-gated modeling for joint slot filling and intent prediction. In Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 2 (Short Papers), pages 753–757
6. Chenwei Zhang, Yaliang Li, Nan Du, Wei Fan, and Philip Yu. 2019. Joint Slot Filling and Intent Detection via Capsule Neural Networks. In Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics. Association for Computational Linguistics, Florence, Italy, 5259–5267.
7. Henry Weld, Guanghao Huang, Jean Lee, Tongshu Zhang, Kunze Wang, Xinghong Guo, Siqu Long, Josiah Poon, and Caren Han. 2021. CONDA: a CONTEXTual Dual-Annotated dataset for in-game toxicity understanding and detection. In Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021. Association for Computational Linguistic. 2406–2416.
8. Soyeon Caren Han, Siqu Long, Huichun Li, Henry Weld, and Josiah Poon. 2021. Bi-directional Joint Neural Network for Intent Classification and Slot Filling. In Proc. Interspeech 2021. ISCA, Brno, Czech Republic, 4743–4747.
9. Qian Chen, Zhu Zhuo, and Wen Wang. 2019. BERT for Joint Intent Classification and Slot Filling. arXiv:1902.10909
10. Henry Weld, Xiaoqi Huang, Siqu Long, Josiah Poon, and Soyeon Caren Han. 2022. A survey of joint intent detection and slot filling models in natural language understanding. *ACM Comput. Surv.* Just Accepted (July 2022).

An abstract background featuring overlapping, semi-transparent teal-colored geometric shapes, primarily triangles and polygons, creating a layered, crystalline effect. The shapes are concentrated in the upper half of the image, with some extending towards the right edge.

Q'n'A

16:15 - 16:30

An abstract background featuring overlapping, semi-transparent teal-colored geometric shapes, primarily triangles and polygons, creating a layered, crystalline effect. The shapes are concentrated in the upper half of the frame, with the rest of the background being plain white.

Thank you!!!