AIDC

INTEL AI DEVCON 2018

AIDC
INTEL AI DEVCON 2018

USING DEEP LEARNING FOR ENTITY DETECTION AND INTENT EXTRACTION IN NATURAL LANGUAGE

Peter Izsak
Deep Learning Data Scientist
NLU Israel, AI Products Group, Intel

intel AI

DEMO PACKAGE AND NLP ARCHITECT

```
Download/wget http://tiny.cc/devcon_ie_ner_demo
unzip devcon_ie_ner_demo

cd demo/
pip install –r install_packages.txt

OR

git clone https://github.com/NervanaSystems/nlp-architect.git
cd nlp-architect
make
. '.nlp_architect_env/bin/activate’
cd ..
```

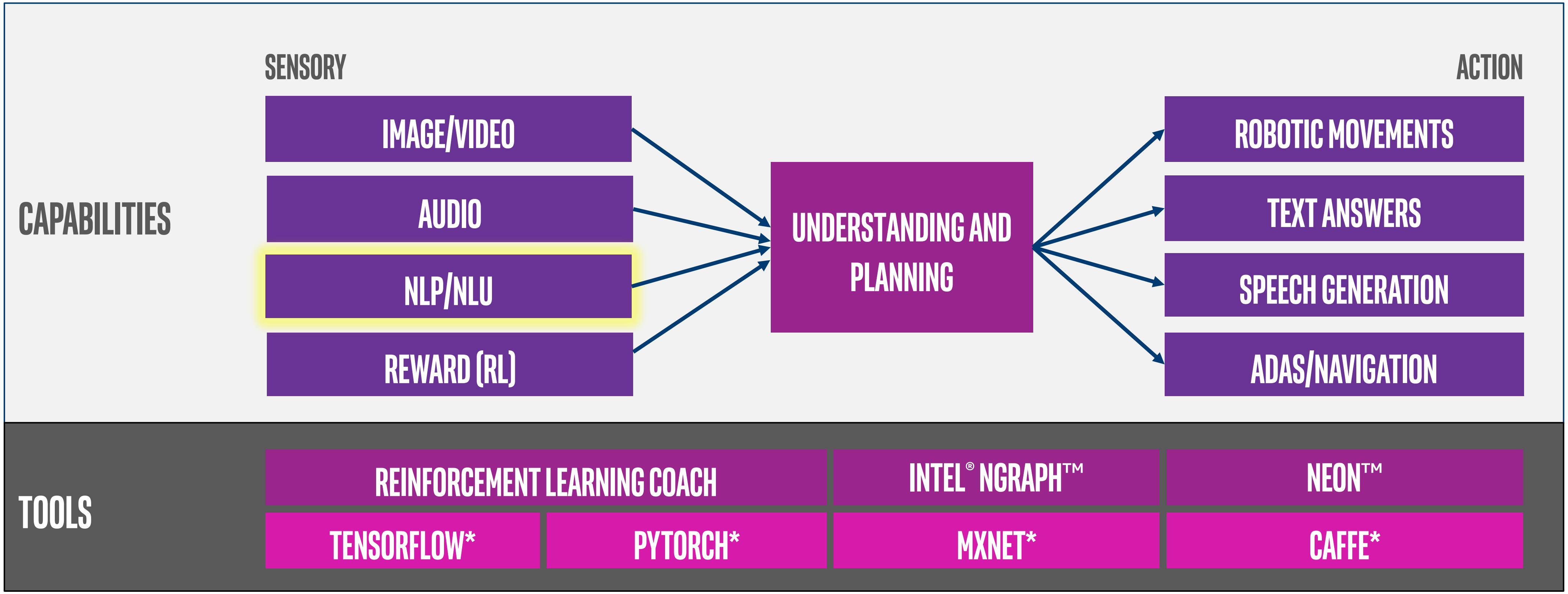
Lunch Jupyter notebook

- Launch demo: jupyter notebook

SESSION OUTLINE

- Intro
- Named Entity Recognition (NER) and Intent Extraction (IE)
- Sequential tagging and best practices
- Model building hands-on
- Application demo with a trained model

INTEL® AI LAB

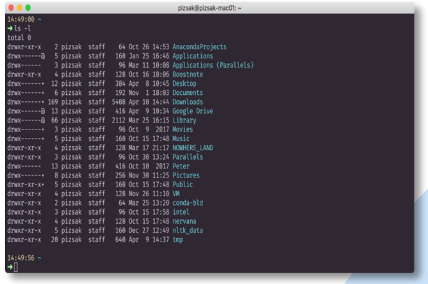


*Other names and brands may be claimed as the property of others.

NLP ARCHITECT BY INTEL® AI LAB

	Python			REST service	
End-to-end Examples	Topic Analysis	Sentiment Analysis	NMT	Dialogue System	Relation Extraction
	Machine Reading Comprehension	Text Summarization	Trend Analysis	Q&A	Intent Extraction
NLP Components	NER	Chunking	Parser	LM	Classification
Topologies	Seq2seq	Sequential Tagging	Memory networks	Attention	GAN
	Word embedding	LSTM/GRU	Matched-LSTM	Convolutional	TCN
Frameworks	Intel® nGraph™	neon™	Tensorflow*	Dynet	other libraries

HUMAN-COMPUTER INTERACTION EVOLUTION



Explicit commands interfaces

Graphical interfaces

Natural Human interaction interfaces

NATURAL LANGUAGE UNDERSTANDING

Will you pick up the kids from school on your way home?

I'm not interested in vacuum cleaners

NAMED ENTITY RECOGNITION

"Book me a flight from <u>Tel-Aviv</u> to <u>San Francisco</u> on <u>May 23rd</u> to

LOCATION LOCATION DATE

visit <u>Intel</u>'s <u>AI DevCon conference</u>"

ORG EVENT

- The task of classifying words or phrases into entity groups
- A Named Entity can be a person, location, organization, date, number, etc..
- Usually the first stage of Information Extraction

INTENT EXTRACTION

"Book me a flight from Tel-Aviv to San Francisco on May 23rd to visit Intel's AI DevCon conference"

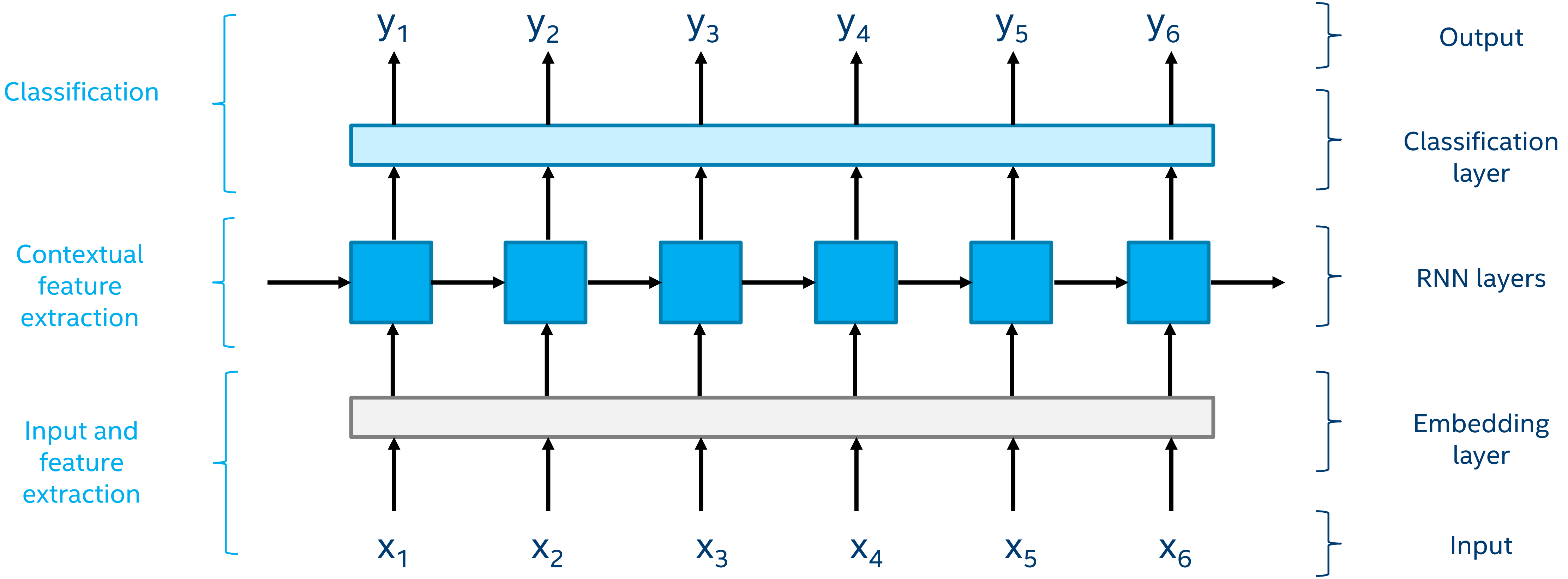
TO-WHOM FROM-CITY TO-CITY DATE

PURPOSE

BOOK FLIGHT INTENT

- The process of understanding commands, requests or promises conveyed in a sentence
- Can be in conjunction with other modalities (or sensors) such as computer vision
- Each intent consists of a set of entities which compose the scenario or command

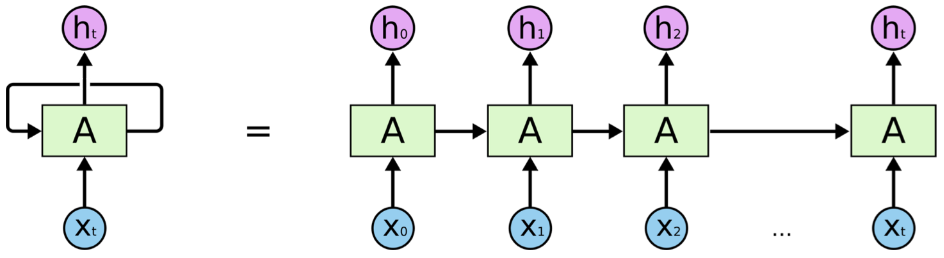
SEQUENTIAL TAGGING USING RECURRENT NEURAL NETWORKS



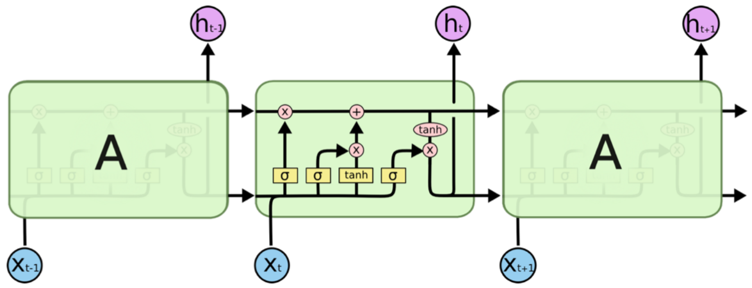
MODELING SEQUENCES WITH RECURRENT NEURAL NETWORKS

- Language data is sequential and has context
- RNNs model temporal input sequences which allows us to model sentences and words
- LSTM/GRU cells are very good at capturing regularities in sequential inputs

RNN cells

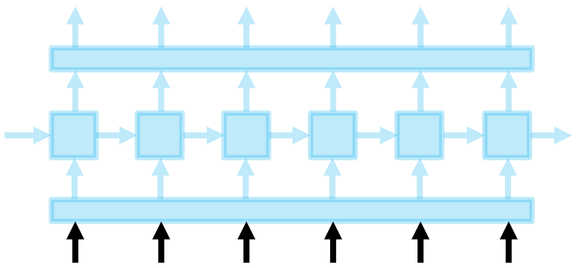


LSTM cells



http://colah.github.io/posts/2015-08-Understanding-LSTMs/

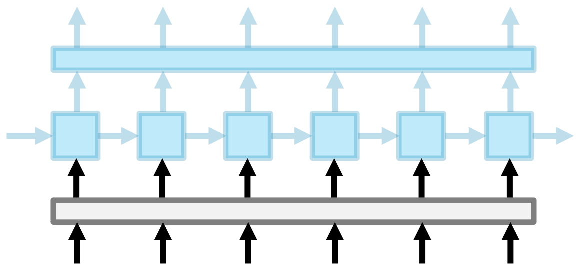
PREPROCESSING STEPS



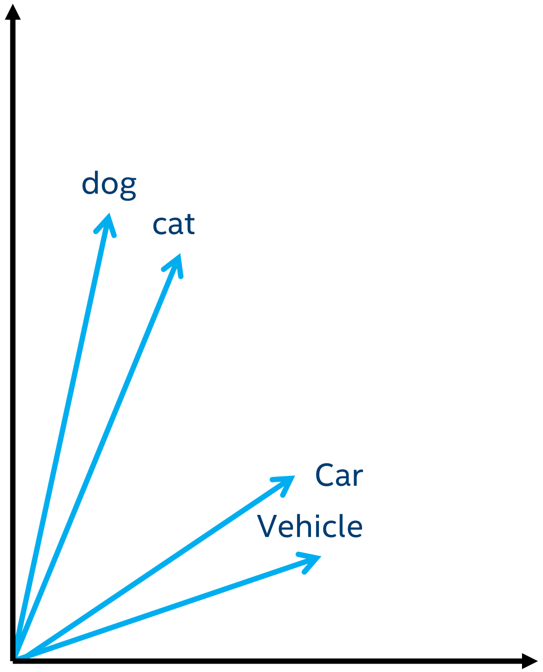
- Textual input can come in various forms
- Preprocessed is necessary
- Common tagging schemes:
  – BIO – mark beginnings with ‘B-’, continue with ‘I-’, other ‘O’
  – IOBES – ‘S’ single tokens, ‘B-’ for label start, ‘I-’ inside and ‘E-’ for end
  – IOB – ‘B-’ to mark new segments of same label

What	is	the	weather	in	California	?
O	O	O	O	O	B-LOC	O

REPRESENTING SENTENCES AND TOKENS

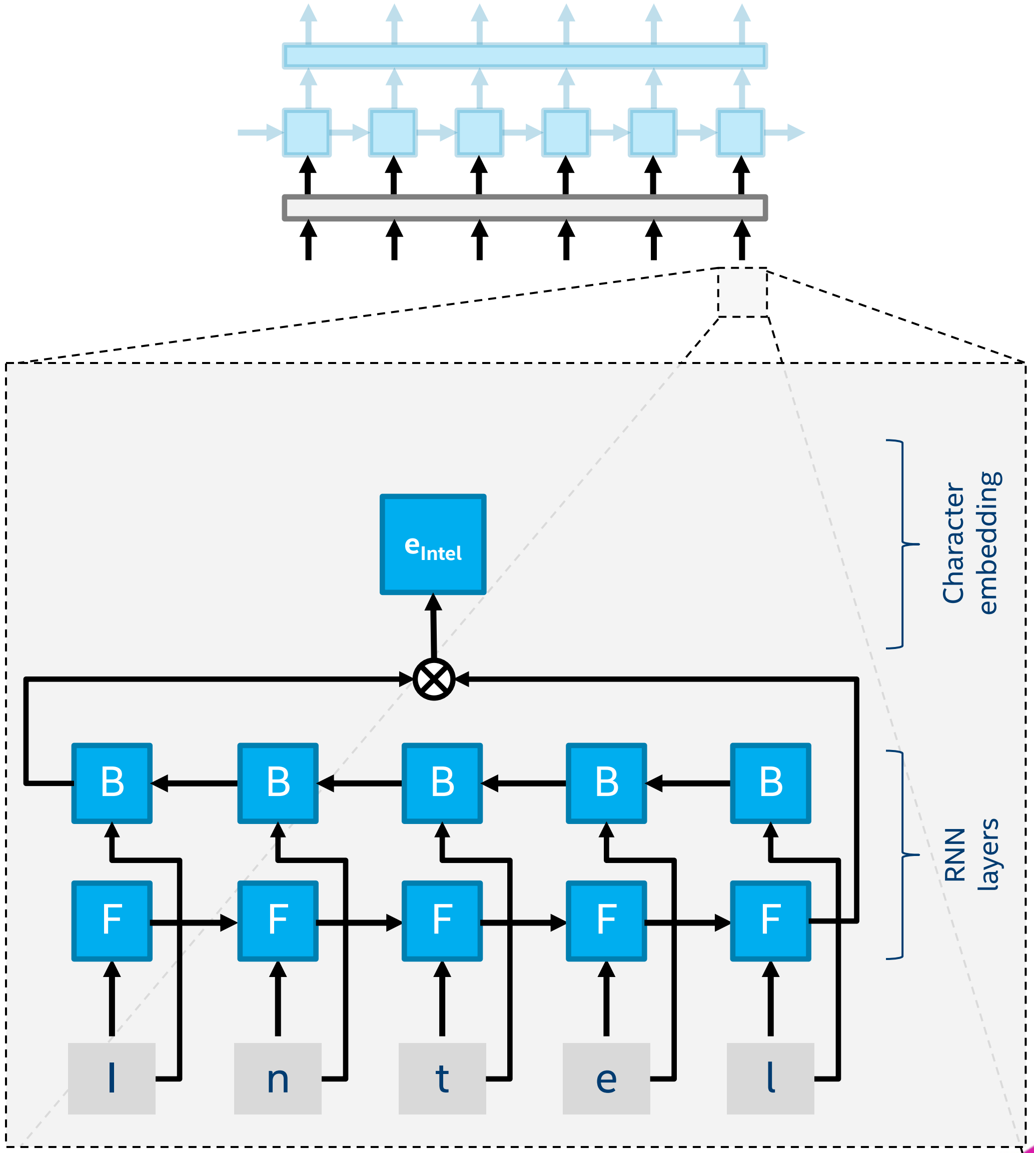


- 1-hot encoding vectors → very sparse
- Dense vector representation (usually <500)
- Word embedding:
  - has boosted the performance of NLP models
  - Can be trained in your model or pre-loaded (Google news w2v/GloVe/Fasttext)

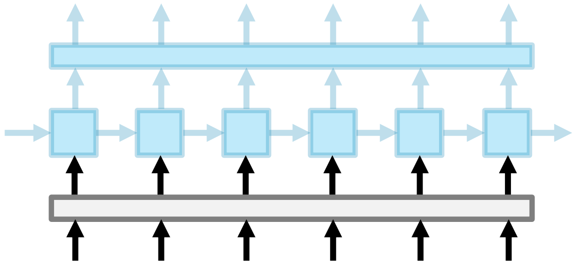


CHARACTER EMBEDDINGS

- Previous DL approaches used pre-built and engineered features:
  - Lexicons
  - Prefix/suffix information
  - Rules
- Learning character embedding
  - Features are learned by the model
  - Domain specific feature extraction

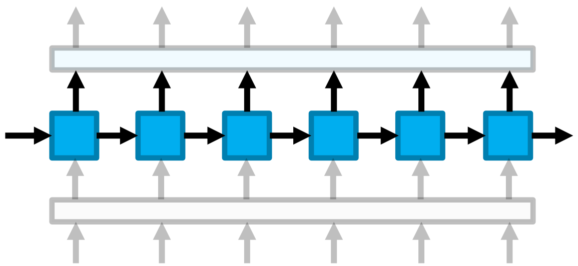


FURTHER OPTIMIZATIONS

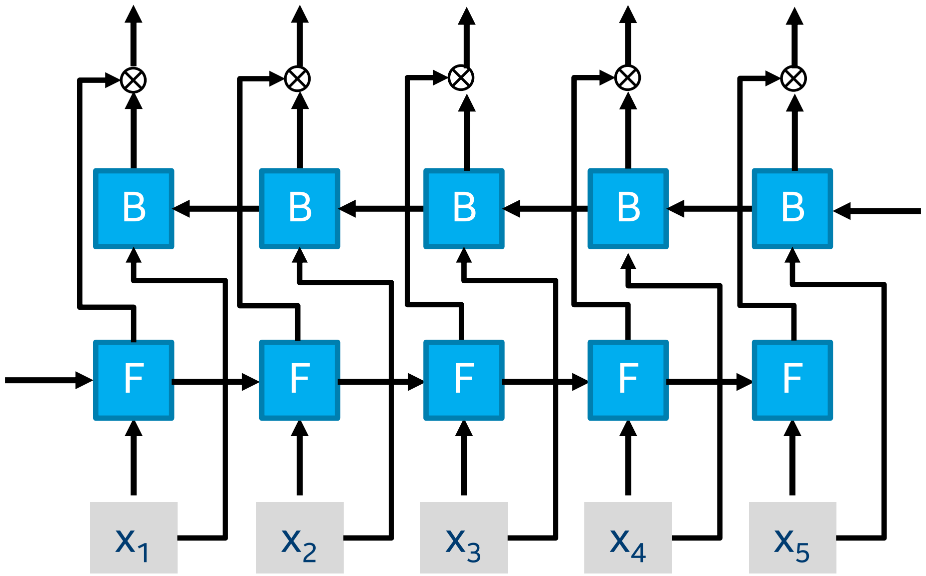


- Different sentences lengths (padding):
  - Add ‘null’ symbols to the right/left-most parts of the sentences
  - Another approach: bucketing
- Handling out-of-vocabulary words:
  - External word embeddings, sub-word embeddings (Fasttext)
  - Character embeddings
  - UNK symbol or heuristic approach

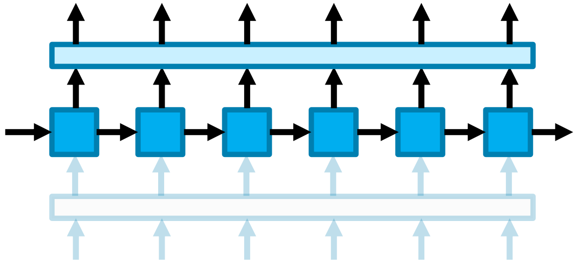
EXTRACTING CONTEXTUAL FEATURES



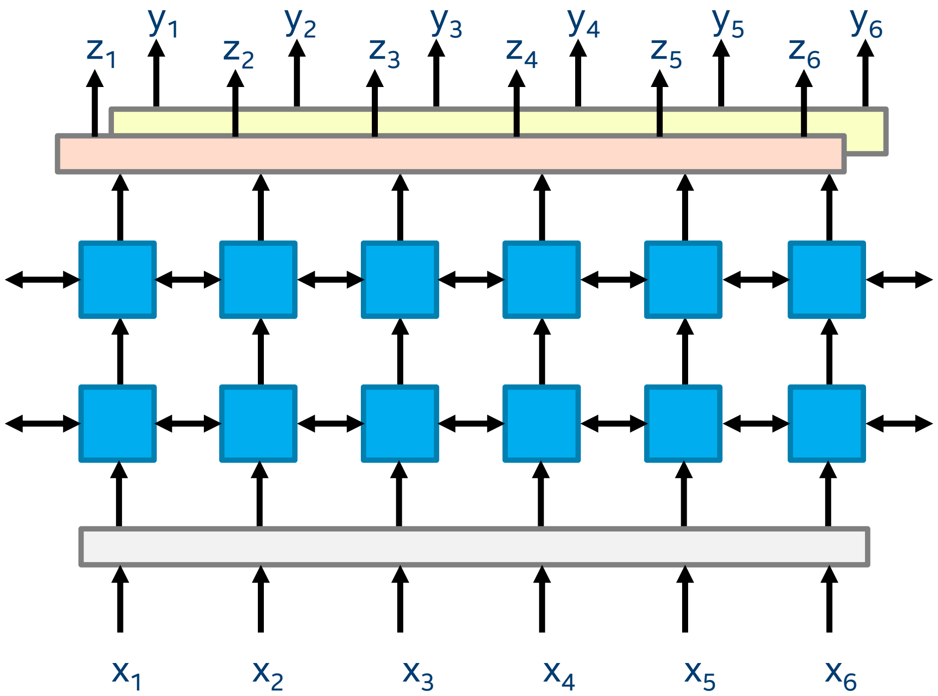
- RNNs are good at modeling the context of sequential data
- Scanning the sentence forward and backward allows classifying tokens based on surrounding windows
- Models using multi-layered Bidirectional LSTM were shown to improve many NLP models



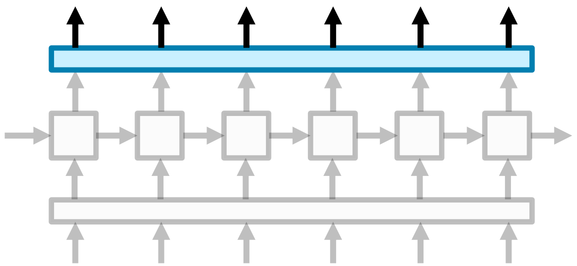
MULTI-TASK LEARNING



- Several related tasks with different goals
- Optimizing one task helps the other tasks
- Networks share underlying embedding or RNN layers
- Classification layers are per task
- Network loss is a weighted sum of per task losses
- Example:
  – Part-of-Speech tagging/Word chunking



PREDICTING THE LABELS



- *Softmax* activation layer
  - Transforms output into a valid probability function over # classes
  - Prediction: $\text{label} = \text{argmax}_{c' \in Labels}(p(c'|y_i))$
  - Individual prediction per token
- Linear-chain Conditional Random Field (CRF)
  - Optimizes transitions between states (labels) as weights
  - Slot classification is done for complete sequences
  - All possible classification combinations should be checked → not very efficient
  - Viterbi algorithm helps for efficient sequence tag decoding

BUILDING A MODEL HANDS-ON

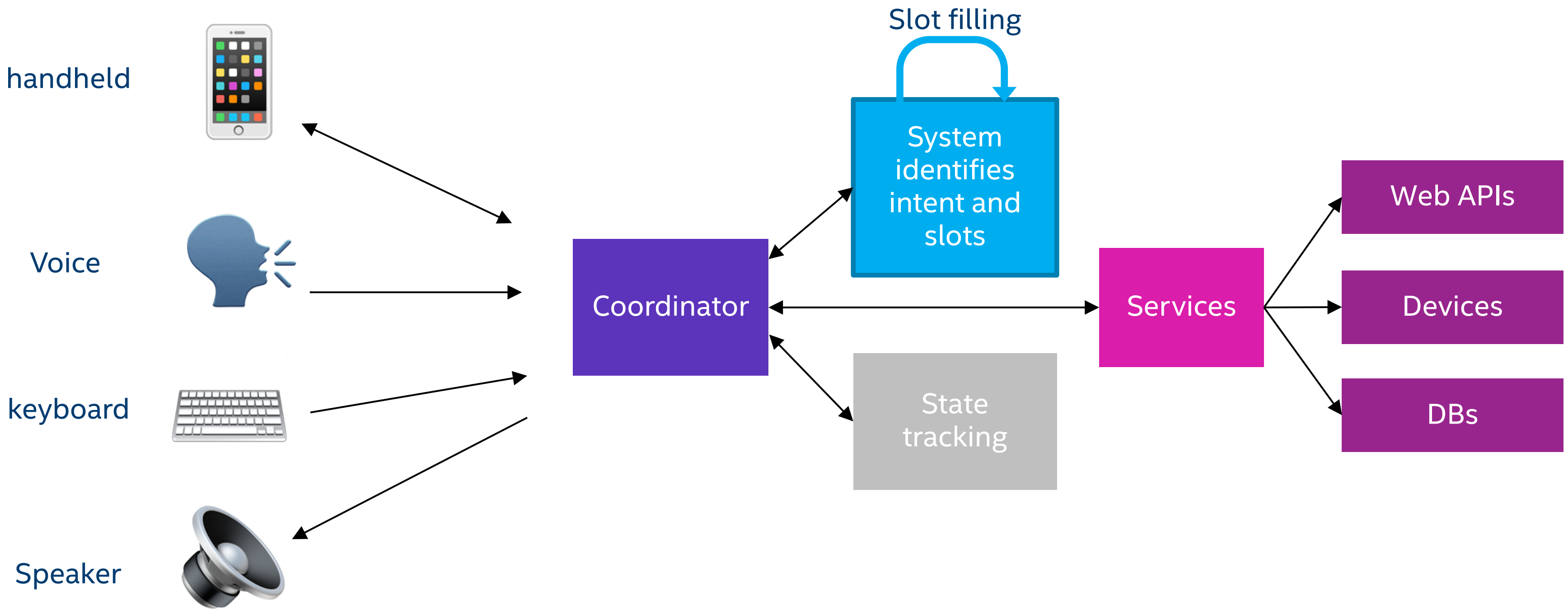
DEMO PACKAGE AND NLP ARCHITECT

```
Download/wget http://tiny.cc/devcon_ie_ner_demo
unzip devcon_ie_ner_demo

cd demo/
pip install –r install_packages.txt

OR

git clone https://github.com/NervanaSystems/nlp-architect.git
cd nlp-architect
make
. '.nlp_architect_env/bin/activate’
cd ..
```

Lunch Jupyter notebook

- Launch demo: jupyter notebook

AN EXAMPLE OF A CONVERSATIONAL APP



REST SERVER/WEB PAGE DEMO

web.py

- Falcon webserver
- Loads model lexicons
- Loads saved model weights
- Create `/intent` endpoint for accepting POST requests

webui/demo.html

- Interactive frontend

SESSION SUMMARY

- Familiarization with NER and IE
- Overview of several best practices for sequential tagging models
- Building an Intent Extraction model hands-on
- Deploying a toy web app using a trained model
- Try out NLP Architect

github.com/NervanaSystems/nlp-architect

LEGAL NOTICES AND DISCLAIMERS

- Intel technologies' features and benefits depend on system configuration and may require enabled hardware, software or service activation. Performance varies depending on system configuration. No computer system can be absolutely secure. Check with your system manufacturer or retailer or learn more at intel.com.
- Software and workloads used in performance tests may have been optimized for performance only on Intel microprocessors. Performance tests, such as SYSmark and MobileMark, are measured using specific computer systems, components, software, operations and functions. Any change to any of those factors may cause the results to vary. You should consult other information and performance tests to assist you in fully evaluating your contemplated purchases, including the performance of that product when combined with other products. For more complete information visit http://www.intel.com/benchmarks.
- Tests document performance of components on a particular test, in specific systems. Differences in hardware, software, or configuration will affect actual performance. Consult other sources of information to evaluate performance as you consider your purchase. For more complete information about performance and benchmark results, visit http://www.intel.com/benchmarks.
- This document contains information on products, services and/or processes in development. All information provided here is subject to change without notice. Contact your Intel representative to obtain the latest forecast, schedule, specifications and roadmaps.
- No license (express or implied, by estoppel or otherwise) to any intellectual property rights is granted by this document.
- Statements in this document that refer to Intel's plans and expectations for the quarter, the year, and the future, are forward-looking statements that involve a number of risks and uncertainties. A detailed discussion of the factors that could affect Intel's results and plans is included in Intel's SEC filings, including the annual report on Form 10-K.
- All products, computer systems, dates and figures specified are preliminary based on current expectations, and are subject to change without notice. The products described may contain design defects or errors known as errata which may cause the product to deviate from published specifications. Current characterized errata are available on request.

LEGAL NOTICES AND DISCLAIMERS

intel® AI

REFERENCES AND RESOURCES

- NLP Architect repository: https://github.com/NervanaSystems/nlp-architect
- Neural network methods for Natural Language Processing, Yoav Goldberg, 2017
- Optimal Hyperparameters for Deep LSTM-Networks for Sequence Labeling Tasks, Reimers et al., 2017, https://arxiv.org/abs/1707.06799
- Neural Architectures for Named Entity Recognition, Lample et al., 2016. https://arxiv.org/abs/1603.01360
- Understanding LSTM Networks, http://colah.github.io/posts/2015-08-Understanding-LSTMs/
- Facebook Fasttext, https://github.com/facebookresearch/fastText
- Jeffrey Pennington, Richard Socher, and Christopher D. Manning. 2014. GloVe: Global Vectors for Word Representation
- https://github.com/snipsco/nlu-benchmark
- https://medium.com/snips-ai/benchmarking-natural-language-understanding-systems-google-facebook-microsoft-and-snips-2b8ddcf9fb19