Deep Natural Language Processing for Search and Recommender Systems


Weiwei Guo


Huiji Gao


Jun Shi


Bo Long


Liang Zhang


Bee-Chung Chen

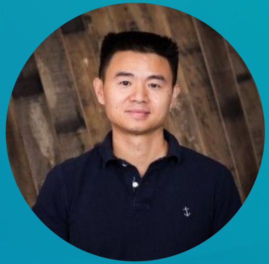

Deepak Agarwal

Agenda

1. Introduction
2. Deep Learning for Natural Language Processing
3. Deep NLP in Search and Recommender Systems
4. Real World Examples

Agenda

1. **Introduction**
2. Deep Learning for Natural Language Processing
3. Deep NLP in Search and Recommender Systems
4. Real World Examples



Introduction

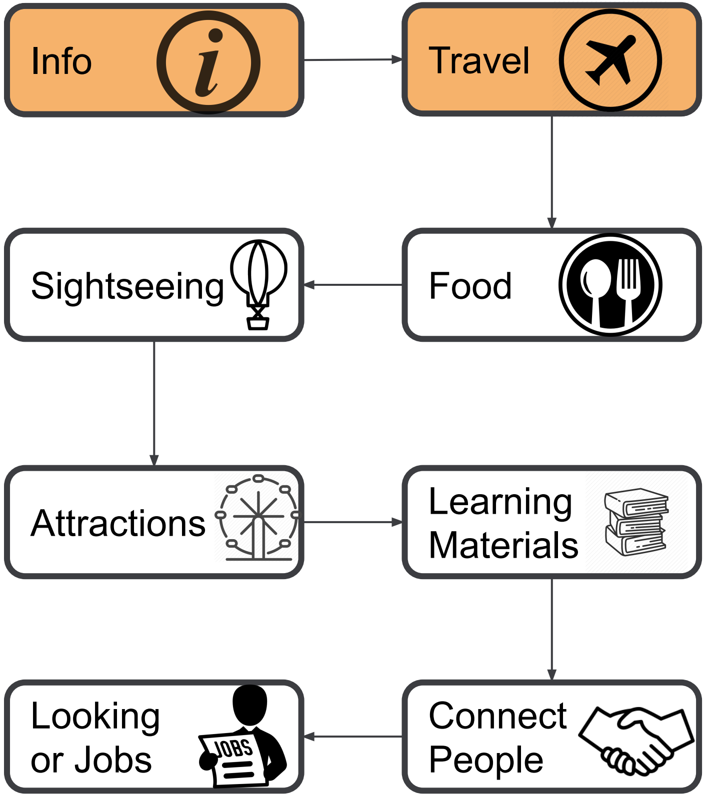
Bo Long

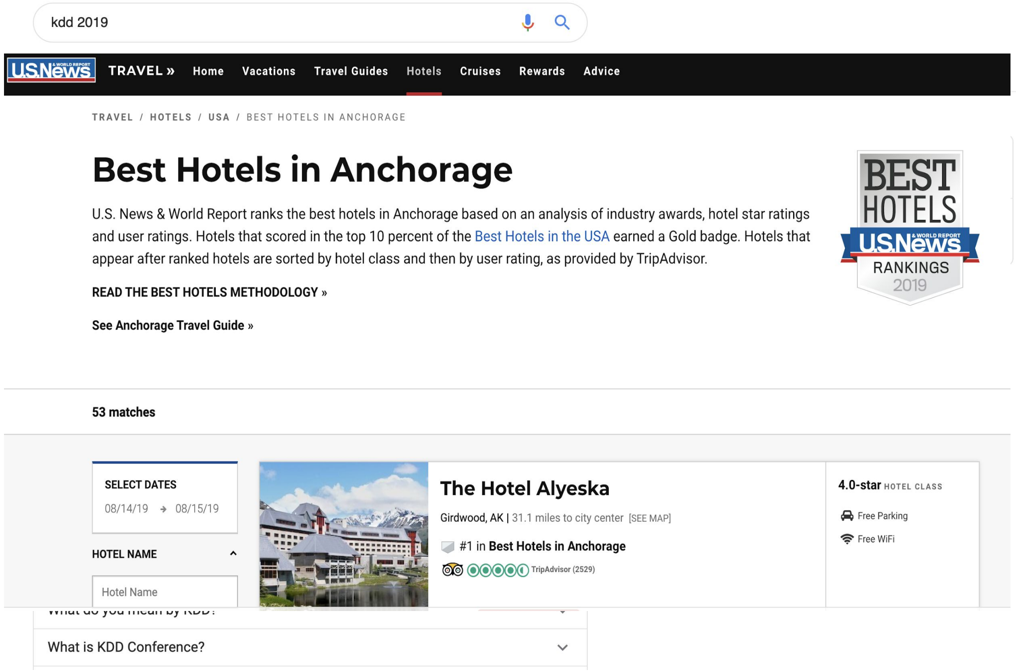
Introduction - Search and Recommender Systems

Search and Recommendation are Everywhere

Introduction - Search and Recommender Systems

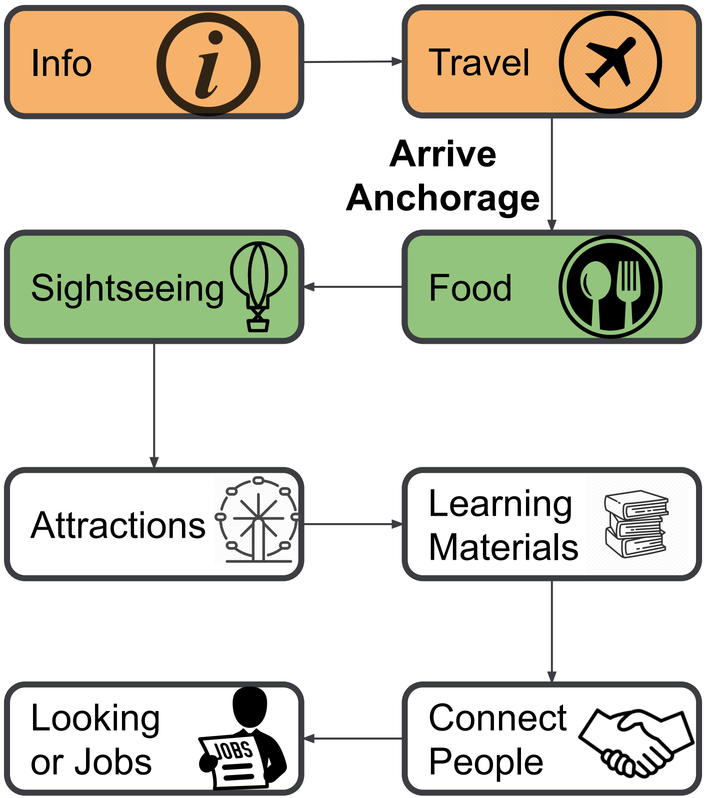
KDD 2019



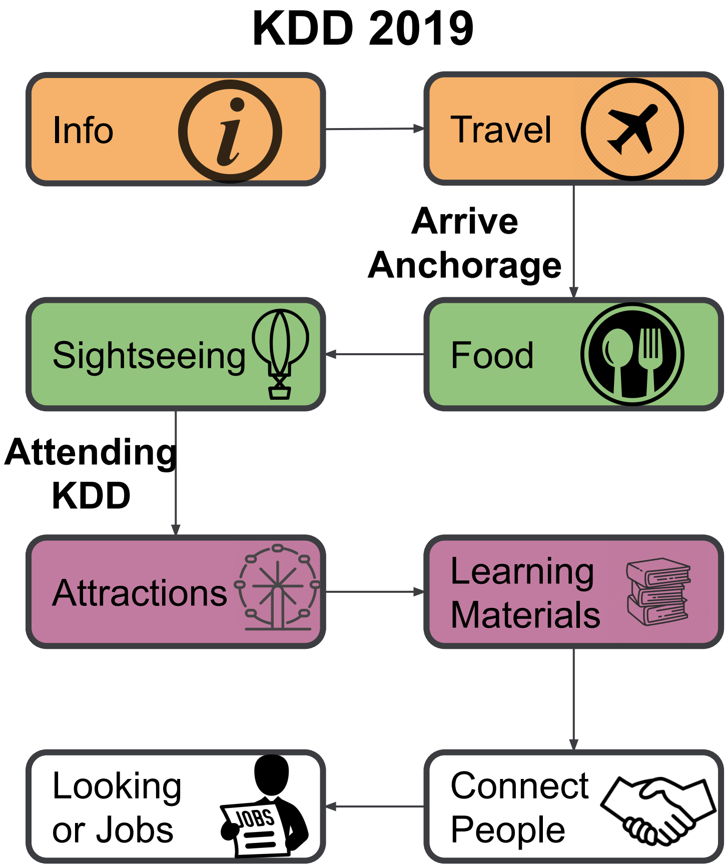


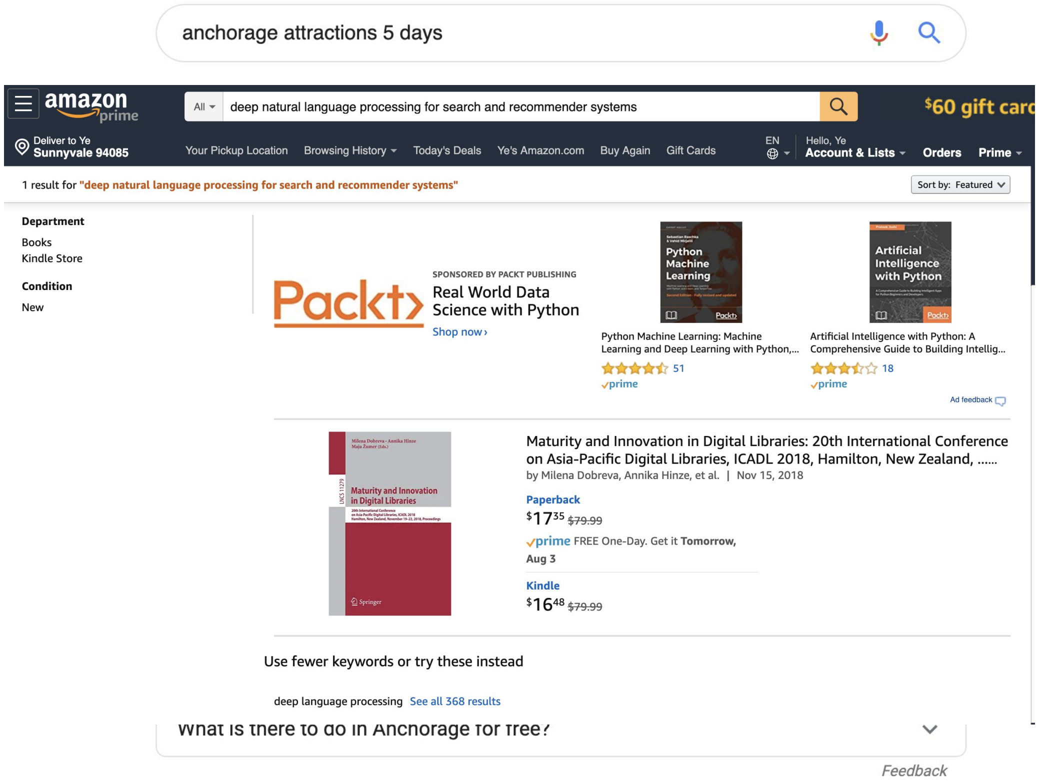
Introduction - Search and Recommender Systems

KDD 2019

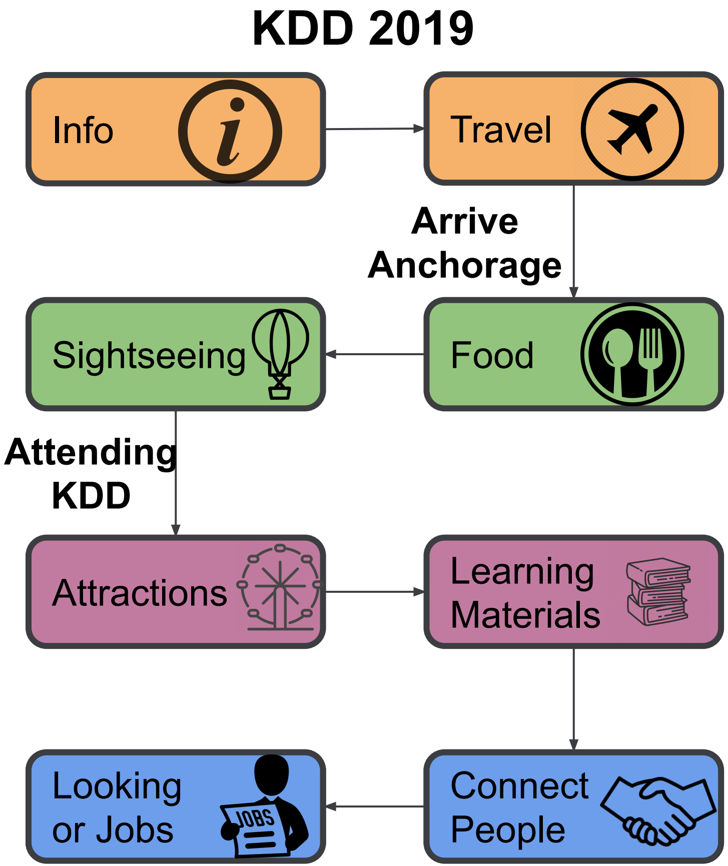


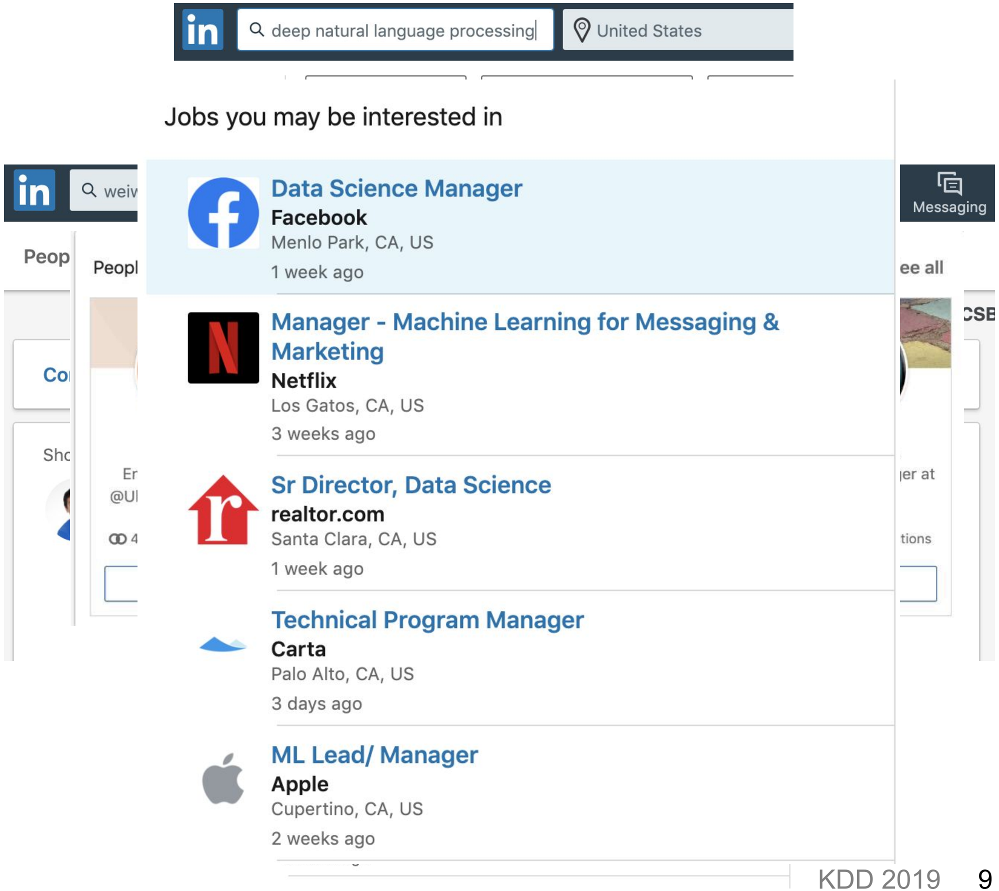
Introduction - Search and Recommender Systems





Introduction - Search and Recommender Systems





Introduction - NLP in Search and Recommender Systems

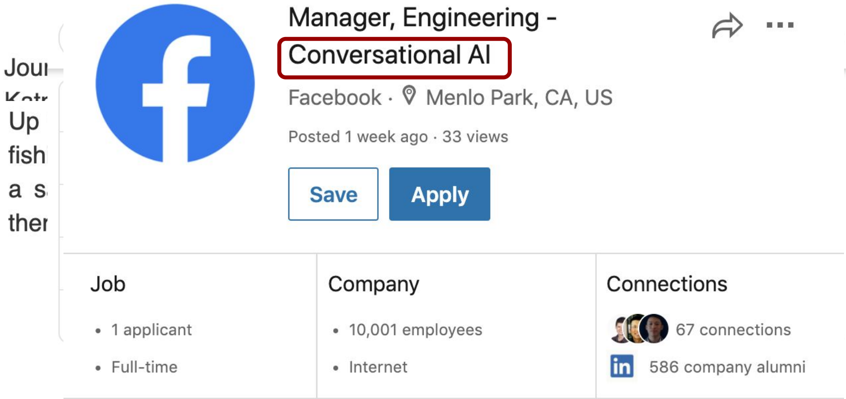
- **Understand User Intention**
  - Search Queries (search)
  - User Profiles (recommendation)
- **Understand Documents**
  - Posts, Reviews, Comments
  - Synonyms, Facets, Semantics, etc.
- **Matching**
  - Query & Doc Matching for Retrieval & Ranking
- **Online Assistance**
  - Query Reformulation

Best Places for Bear Viewing

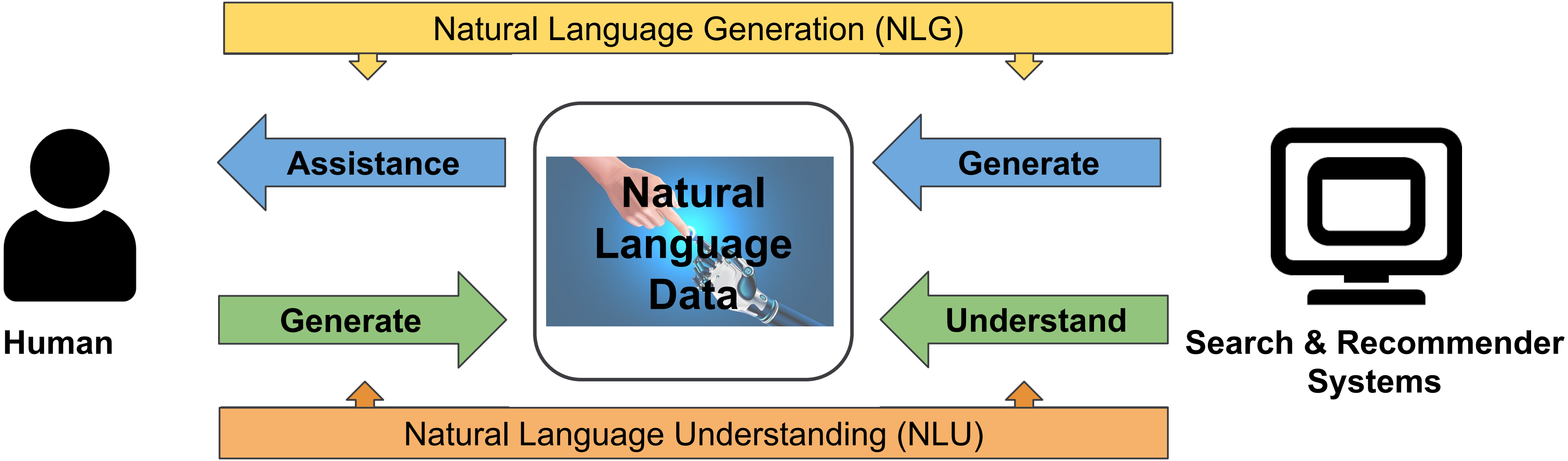
Budget? Location? ...

Engineering Manager - Machine Learning and AI
Aug 2018 – Present · 1 yr 1 mo
San Francisco Bay Area

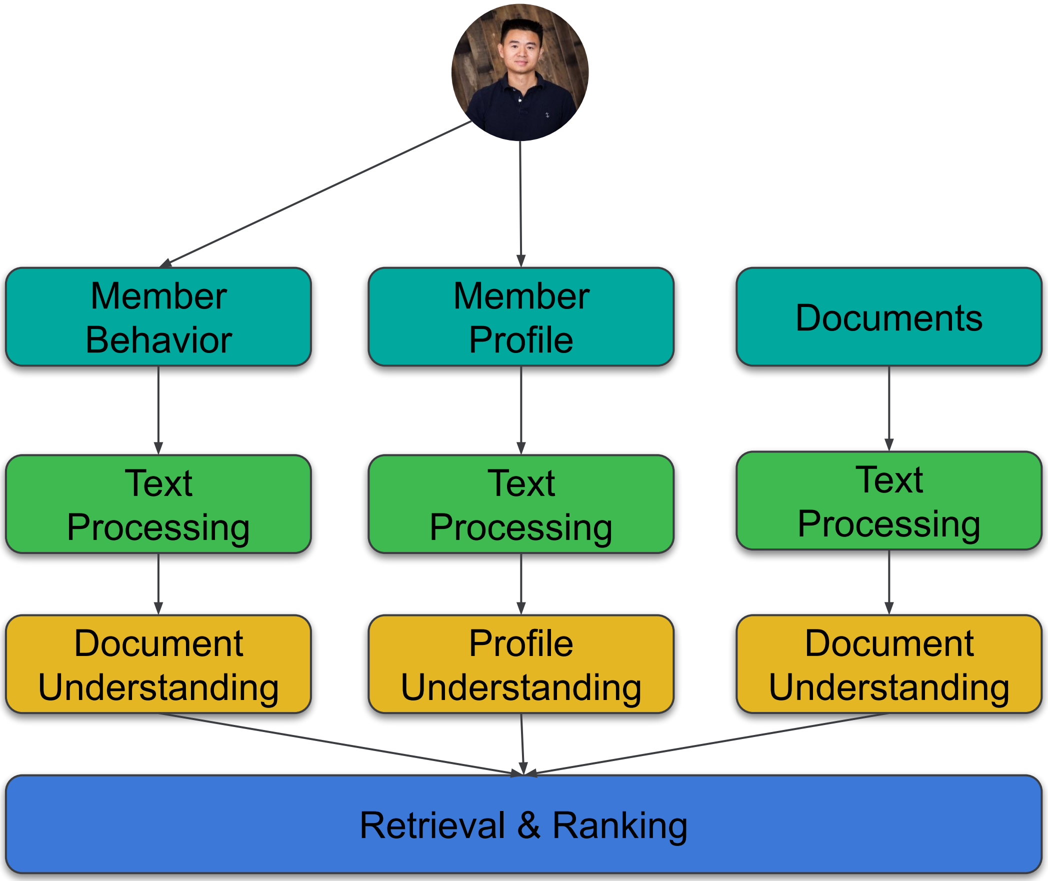
Lead LinkedIn Personalization and Search AI Foundation team - Provide LinkedIn users with intelligent experience through natural language understanding (across multiple languages) and personalization powered by Machine Learning and Artificial Intelligence.



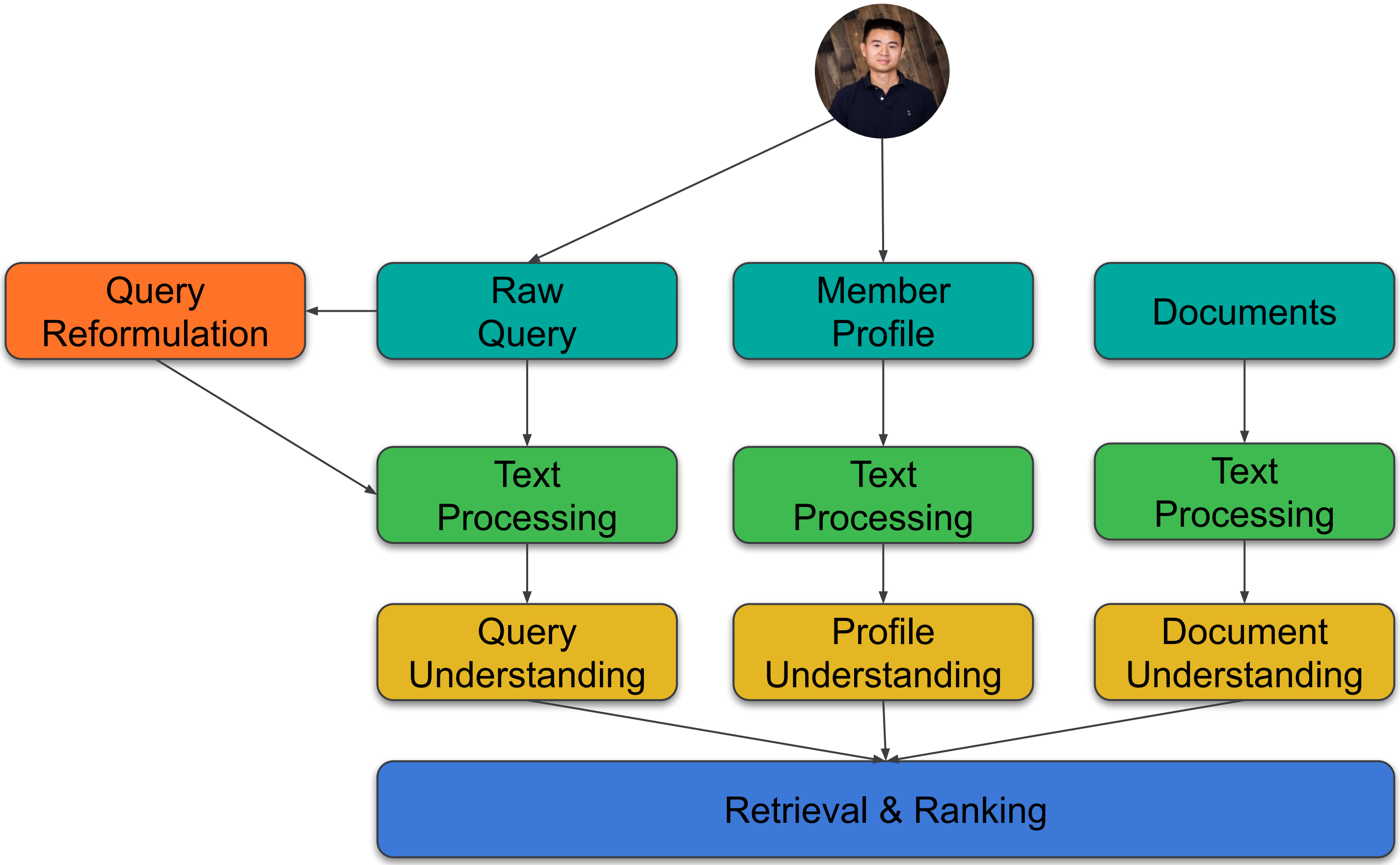
Introduction - NLP in Search and Recommender Systems



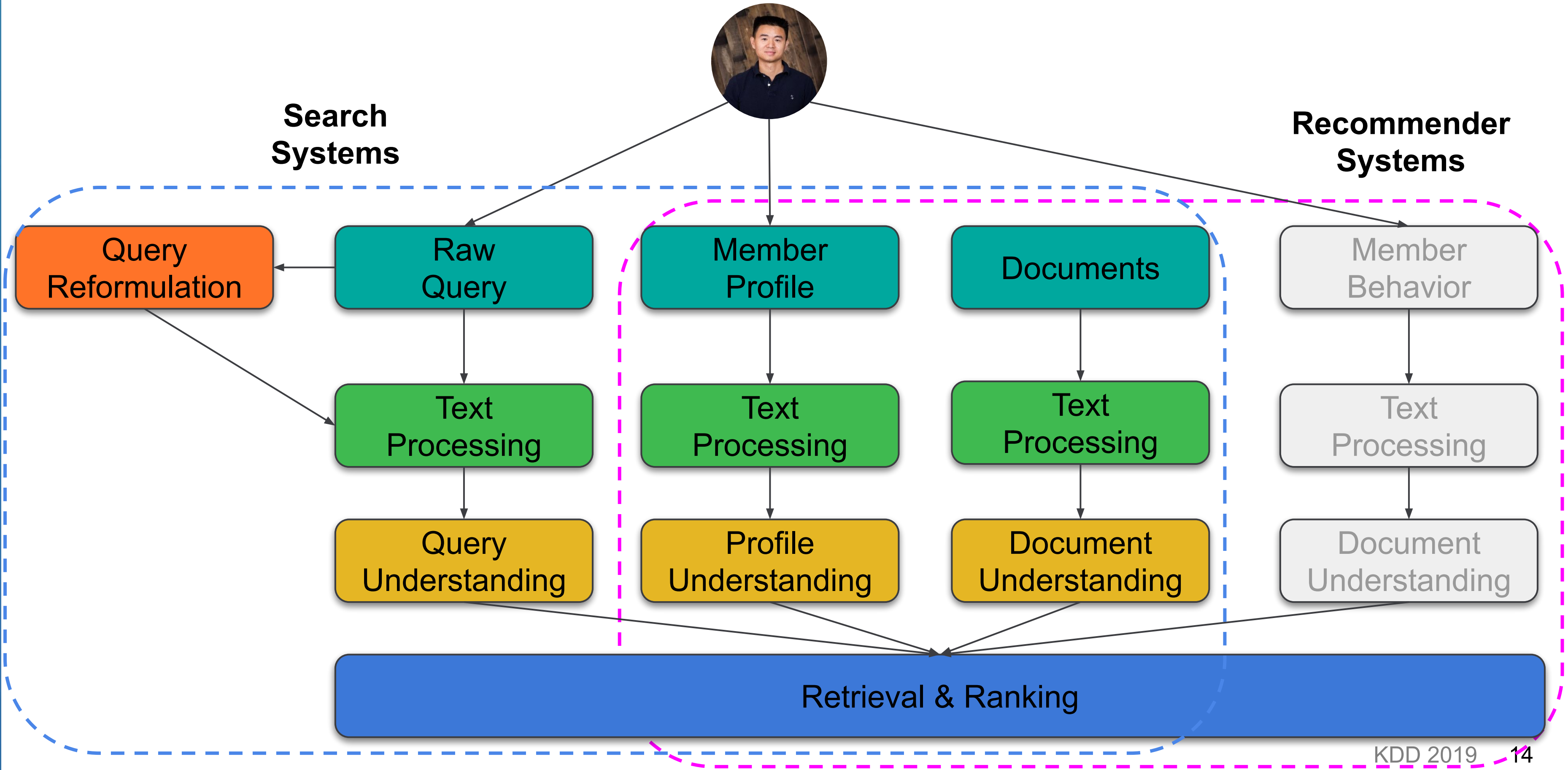
Natural Language Processing in Recommendation Ecosystem



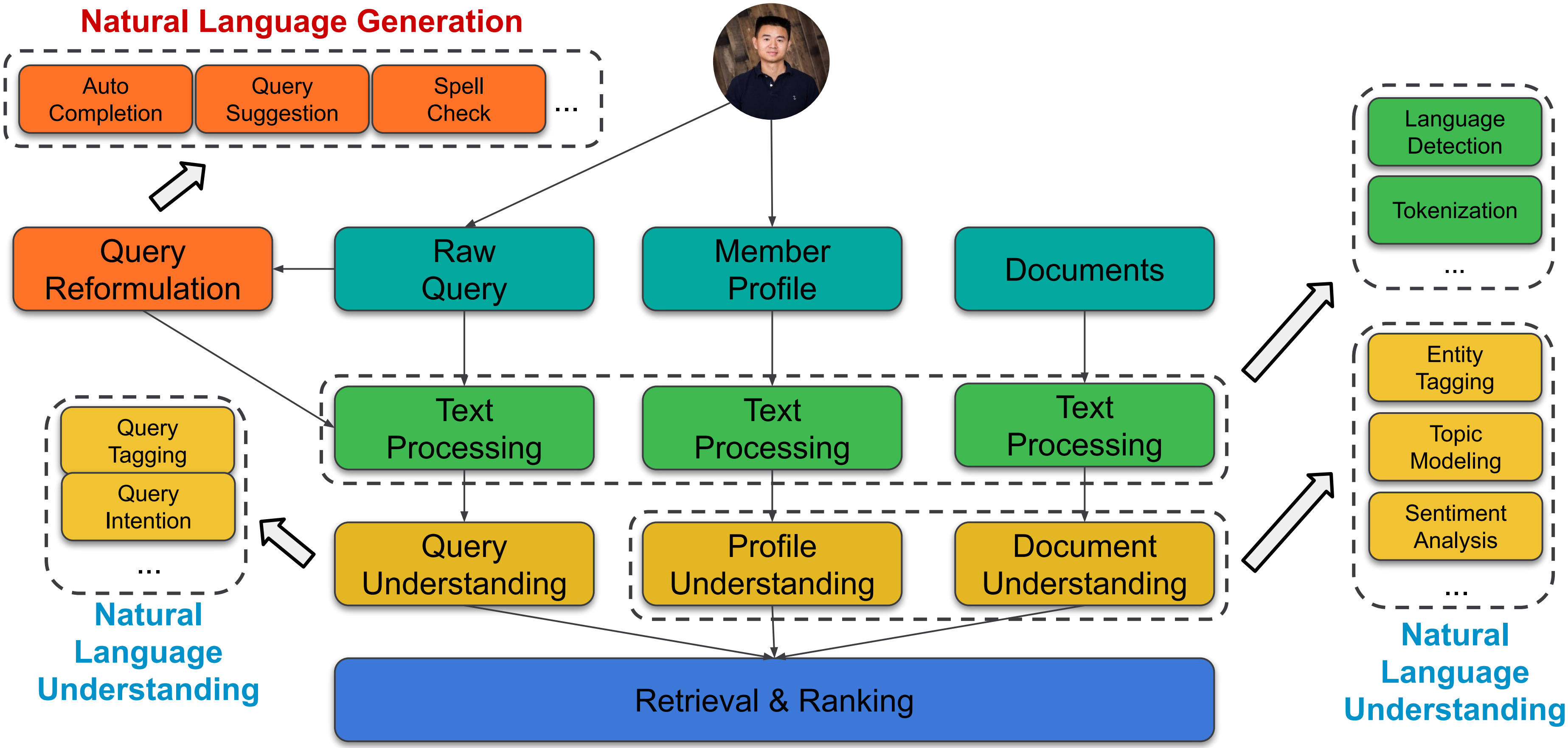
Natural Language Processing in Search Ecosystem



Natural Language Processing in Search and Recommendation



Natural Language Processing in Search and Recommendation



Introduction - Deep Learning for NLP

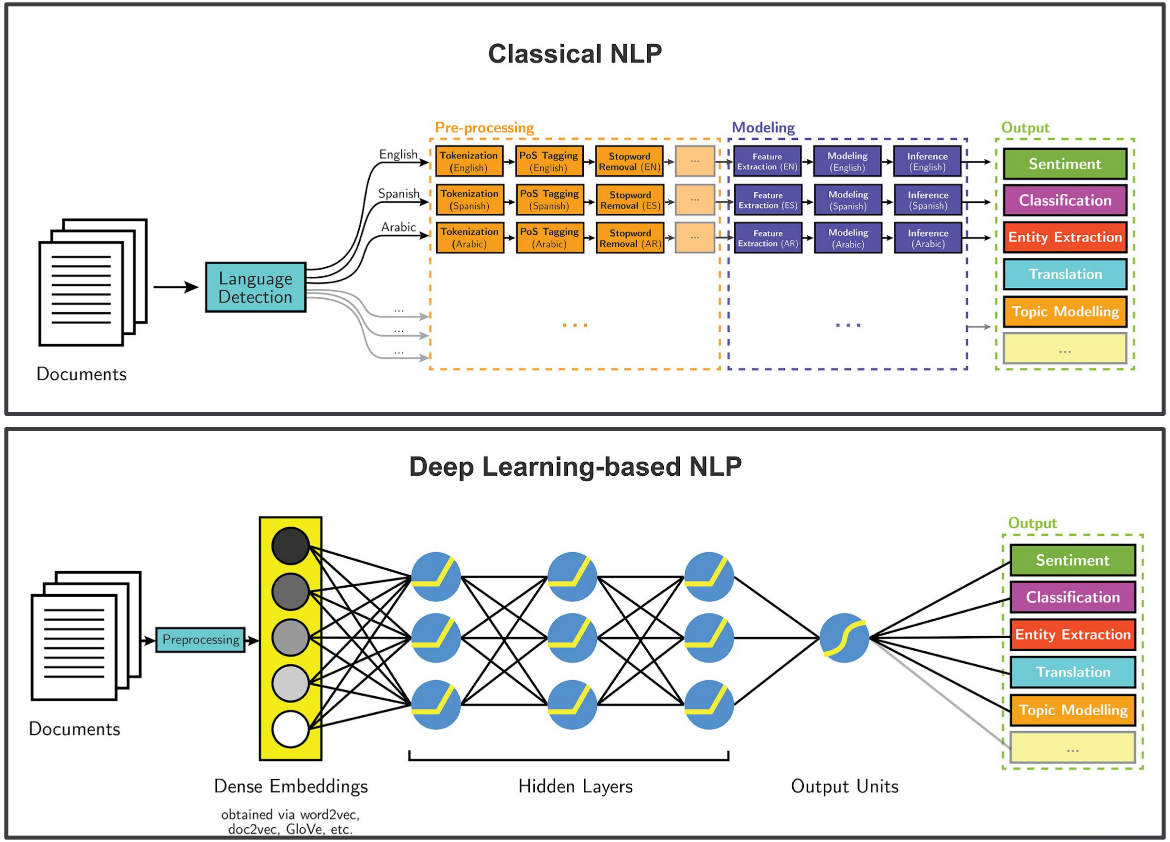


Figure source: Elvis, 2018, https://medium.com/dair-ai/deep-learning-for-nlp-an-overview-of-recent-trends-d0d8f40a776d

Opportunities - Deep Learning for NLP in Search and Recommender Systems

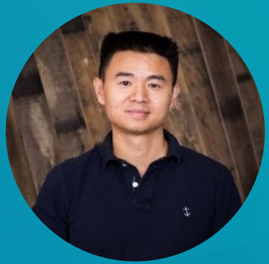
Why Deep Learning?

- **Deep Semantics** from High Dimension and Sparse Data
  - Synonymous, Disambiguation, etc.
- **Easy Feature Engineering**
  - Hand Crafted Features V.S. Auto Feature Representation
- **Model Flexibility**
  - Model end-to-end process
  - Various NN components to model and cooperate systematically
- **Multi-level Feature Representation**
  - Hierarchical Representations
    character -> token -> word -> phrase -> sentence

[Young et. al. 2018]

Agenda

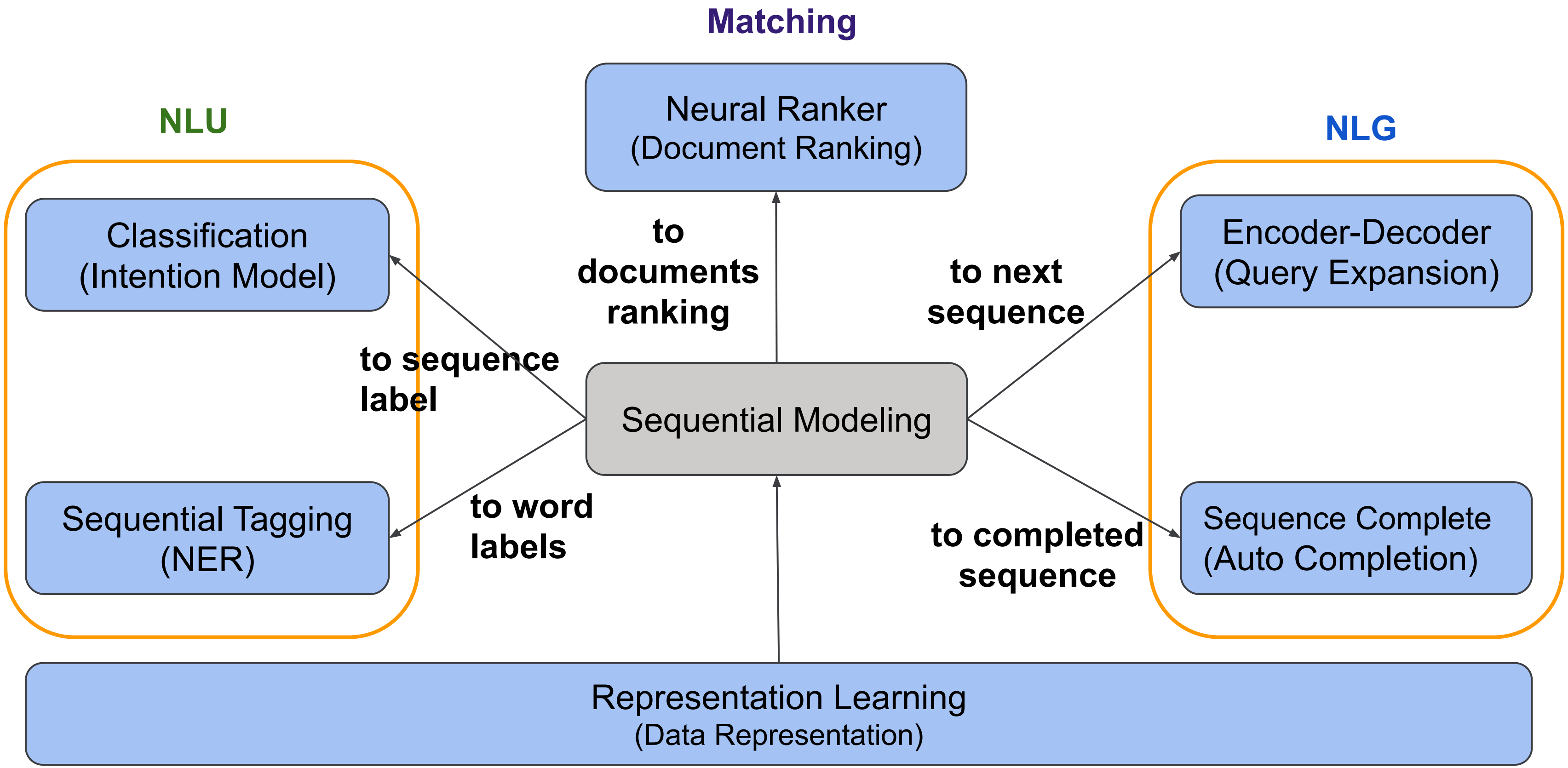
1. Introduction
2. **Deep Learning for Natural Language Processing**
3. Deep NLP in Search and Recommender Systems
4. Real World Examples



Deep Learning for Natural Language Processing

Bo Long

Deep Learning for Natural Language Processing



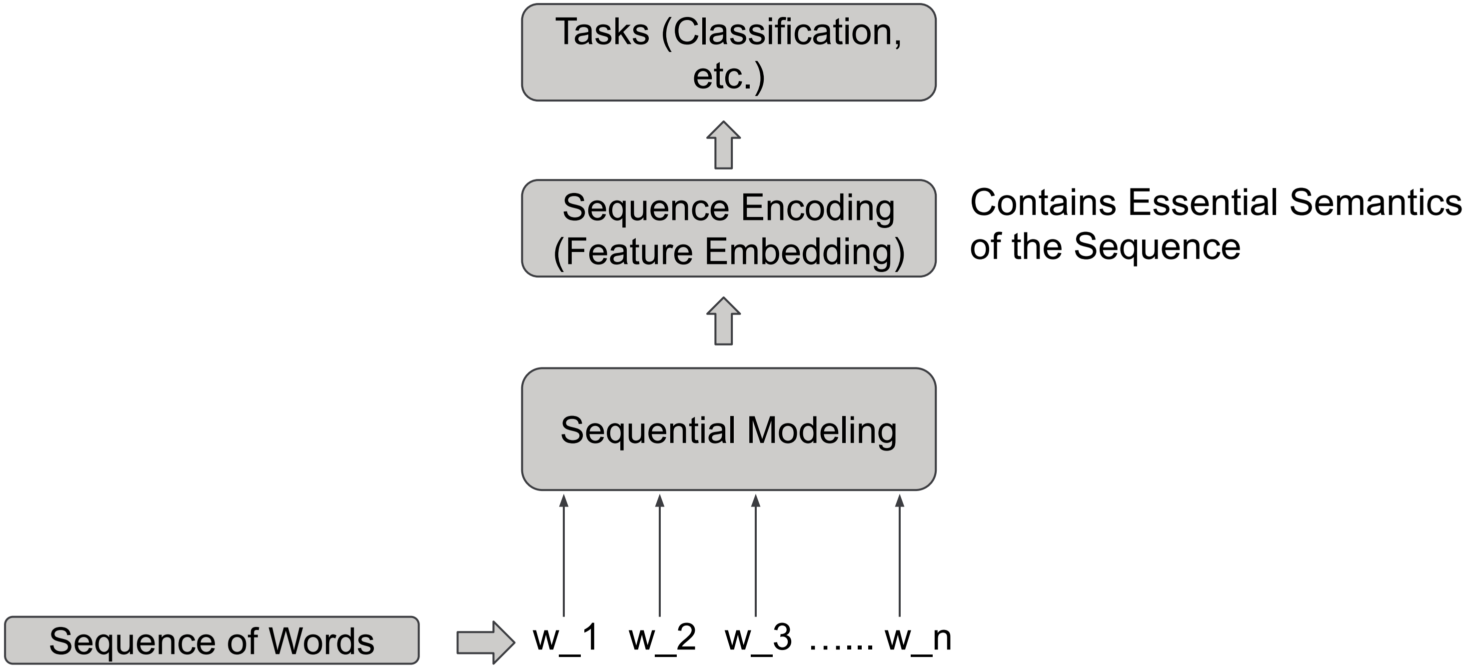
Deep Learning for Natural Language Processing

- **Sequential Modeling on Semantic Context**

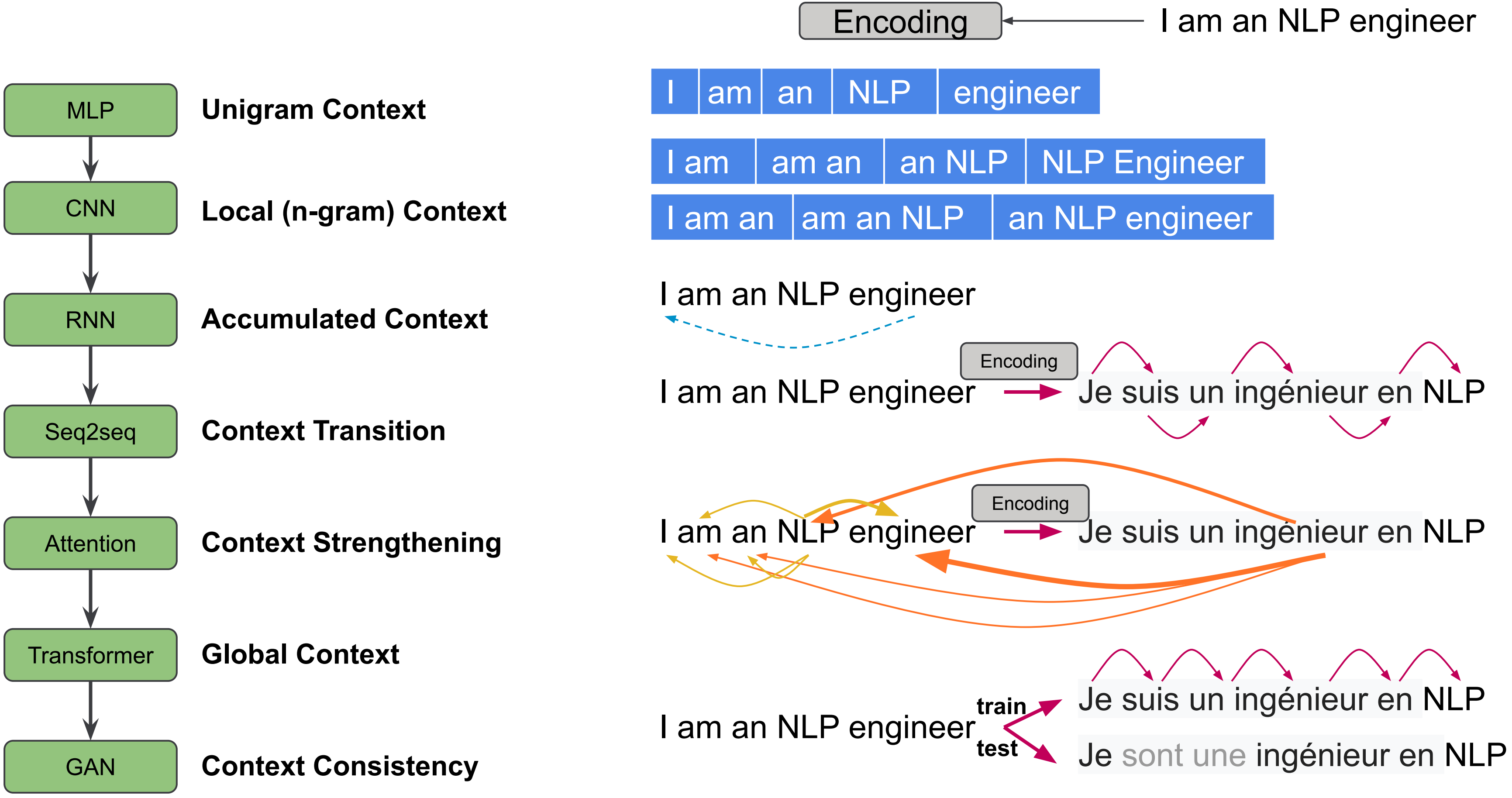
- Representation Learning for Data Processing

Deep Learning for Natural Language Processing

- **Sequential Modeling on Semantic Context**



Sequential Modeling on Semantic Context



Multilayer Perceptron (MLP)

MLP	**Unigram Context**
CNN	Local (n-gram) Context
RNN	Accumulated Context
Seq2seq	Context Transition
Attention	Context Strengthening
Transformer	Global Context
GAN	Context Consistency

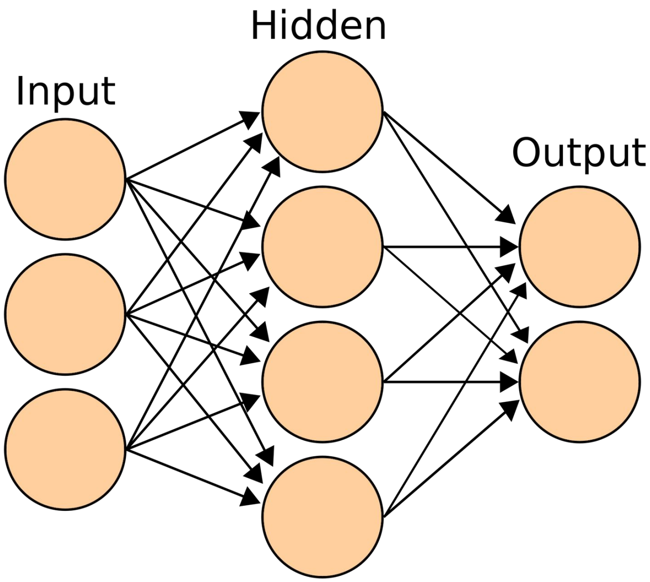


Figure source: https://stackabuse.com/introduction-to-neural-networks-with-scikit-learn/

Convolutional Neural Networks (CNN)

MLP — **Unigram Context**

CNN — **Local (n-gram) Context**

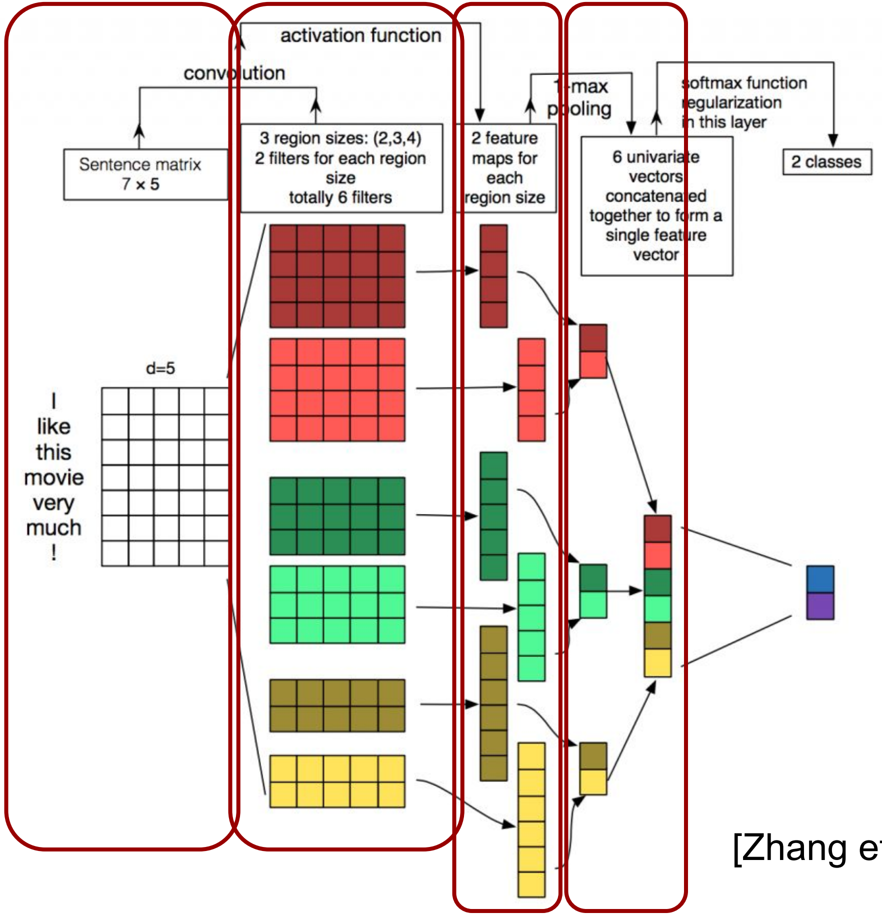
RNN — **Accumulated Context**

Seq2seq — **Context Transition**

Attention — **Context Strengthening**

Transformer — **Global Context**

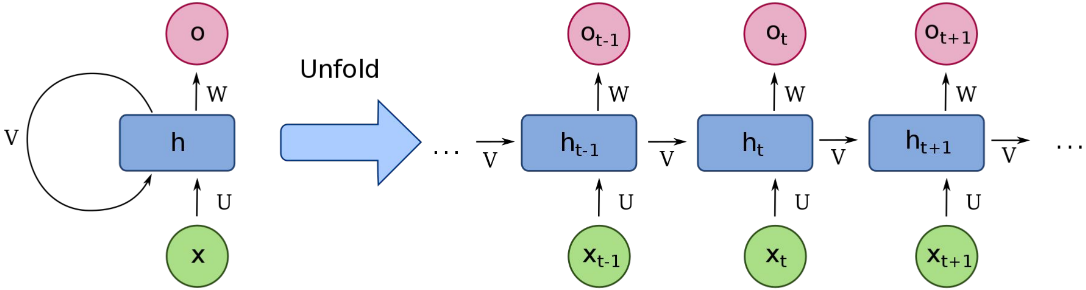
GAN — **Context Consistency**



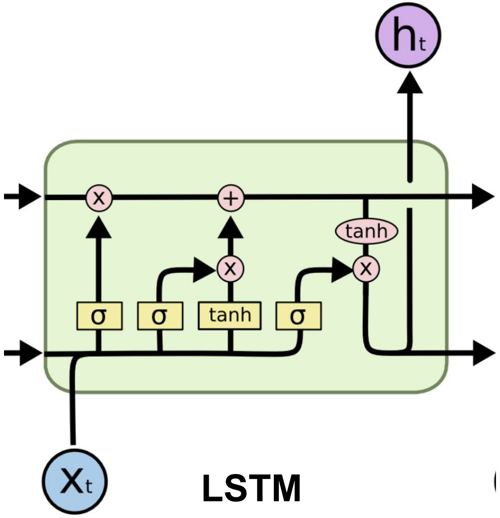
[Zhang et. al. 2015]

Recurrent Neural Networks (RNN)

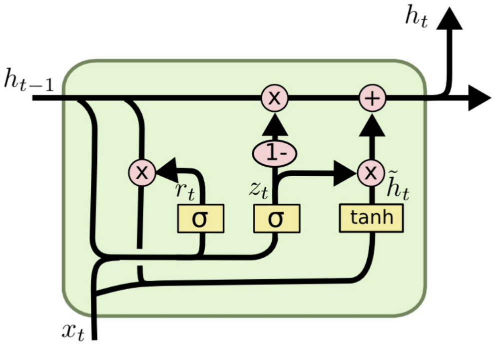
- MLP — **Unigram Context**
- CNN — **Local (n-gram) Context**
- RNN — **Accumulated Context**
- Seq2seq — **Context Transition**
- Attention — **Context Strengthening**
- Transformer — **Global Context**
- GAN — **Context Consistency**



[LeCun et. al. 2015]



LSTM

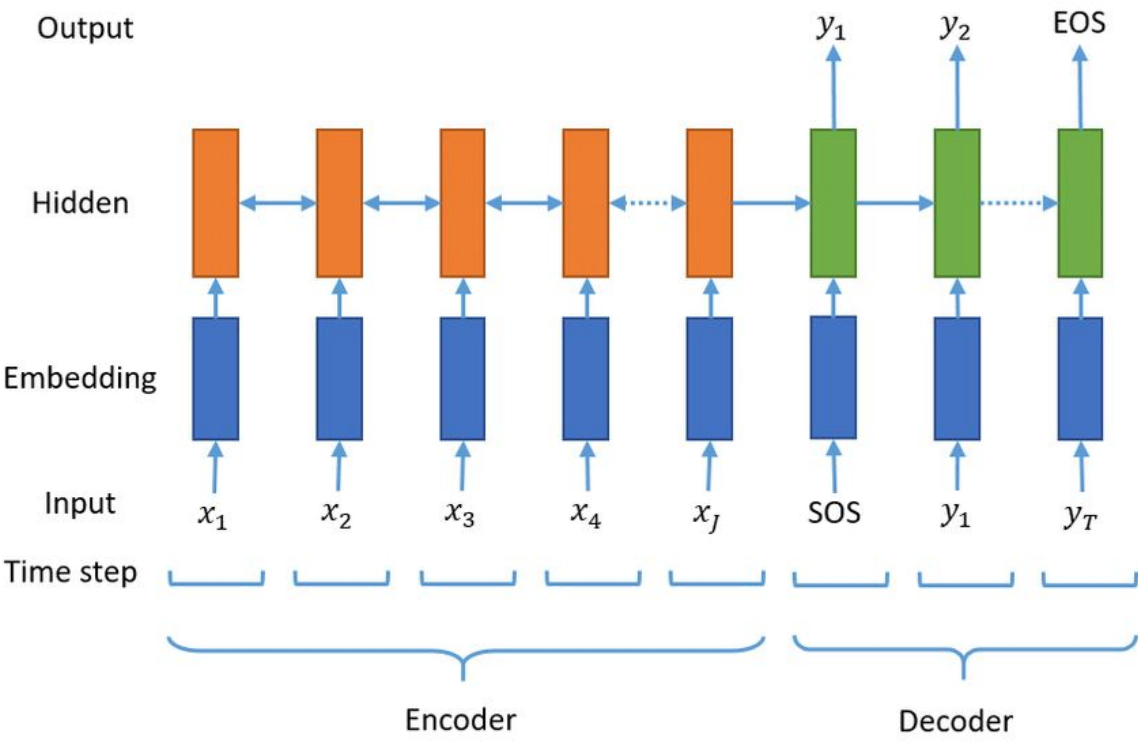


GRU

github: 2015-08-Understanding-LSTMs

Sequence to Sequence (Encoder - Decoder)

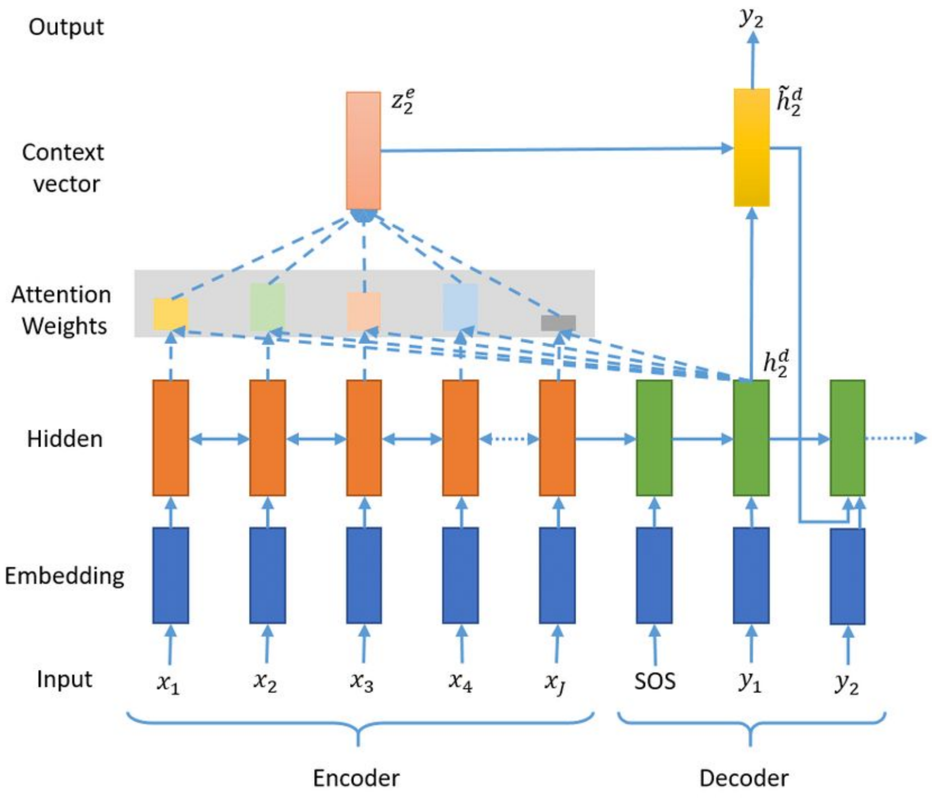
MLP	**Unigram Context**
CNN	**Local (n-gram) Context**
RNN	**Accumulated Context**
Seq2seq	**Context Transition**
Attention	**Context Strengthening**
Transformer	**Global Context**
GAN	**Context Consistency**



[Shi et. al. 2018]

Attention Mechanisms

- MLP — **Unigram Context**
- CNN — **Local (n-gram) Context**
- RNN — **Accumulated Context**
- Seq2seq — **Context Transition**
- Attention — **Context Strengthening**
- Transformer — **Global Context**
- GAN — **Context Consistency**

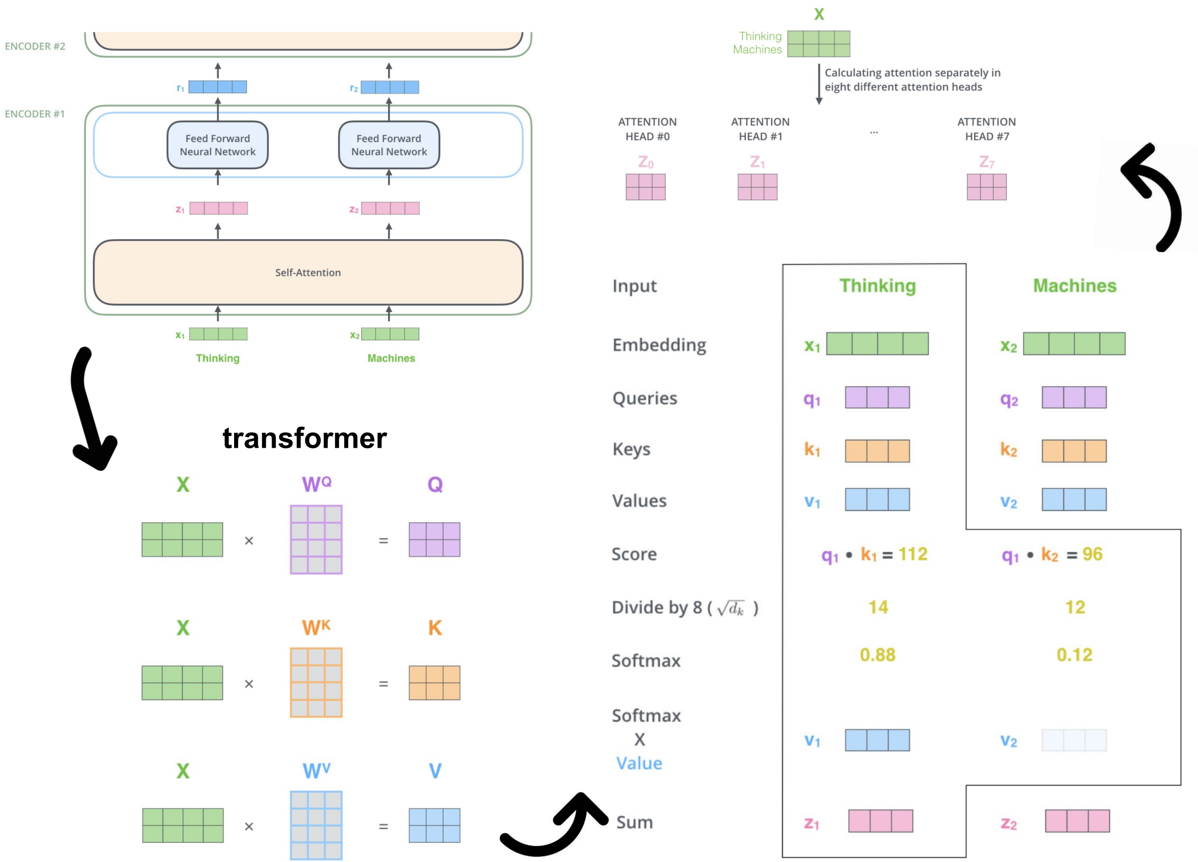


[Shi et. al. 2018]

Transformer

Novelty: Eliminate Recurrence

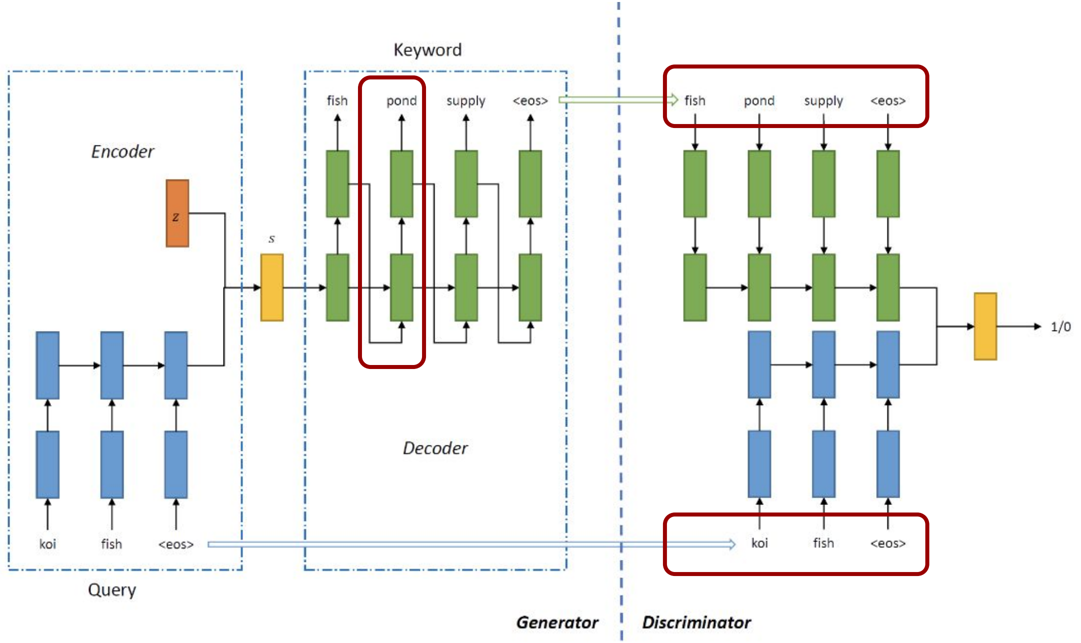
MLP	**Unigram Context**
CNN	**Local (n-gram) Context**
RNN	**Accumulated Context**
Seq2seq	**Context Transition**
Attention	**Context Strengthening**
Transformer	**Global Context**
GAN	**Context Consistency**



http://jalammar.github.io/illustrated-transformer/

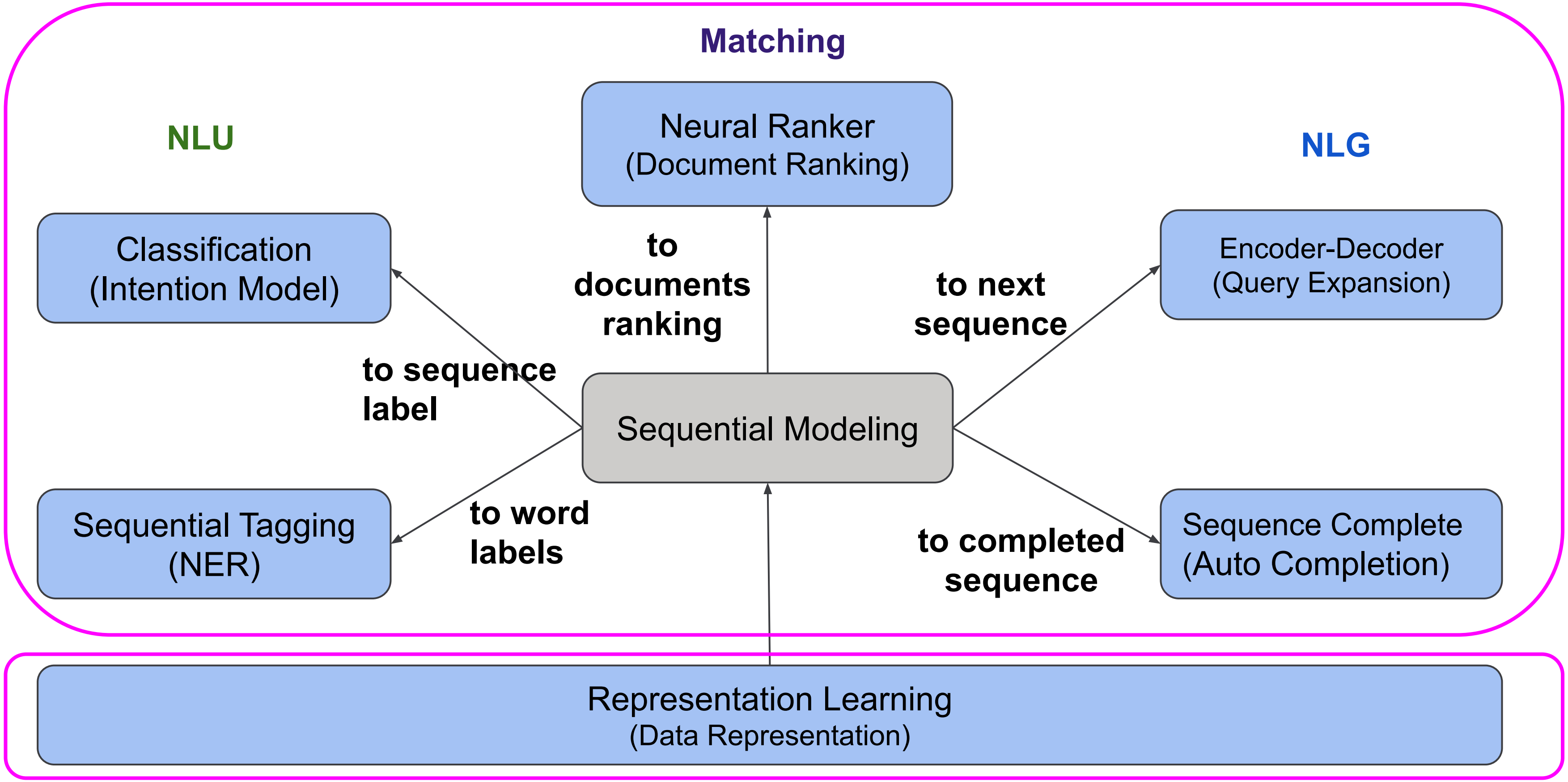
Generative Adversarial Networks (GANs)

MLP	**Unigram Context**
CNN	**Local (n-gram) Context**
RNN	**Accumulated Context**
Seq2seq	**Context Transition**
Attention	**Context Strengthening**
Transformer	**Global Context**
GAN	**Context Consistency**



[Lee, et al., KDD 2018]

Deep Learning for Natural Language Processing



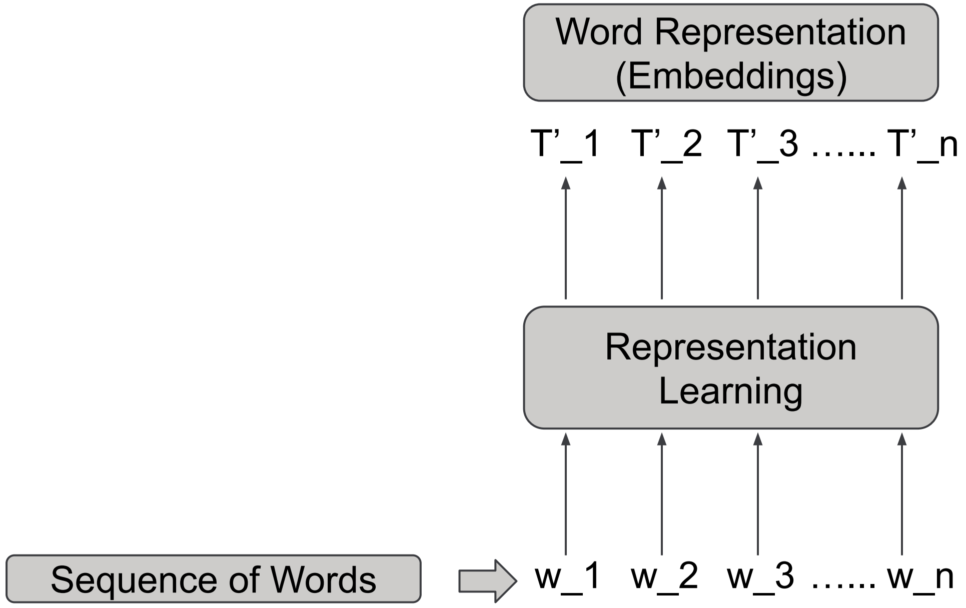
Deep Learning for Natural Language Processing

- Sequential Modeling on Semantic Context

- **Representation Learning for Data Processing**

Deep Learning for Natural Language Processing

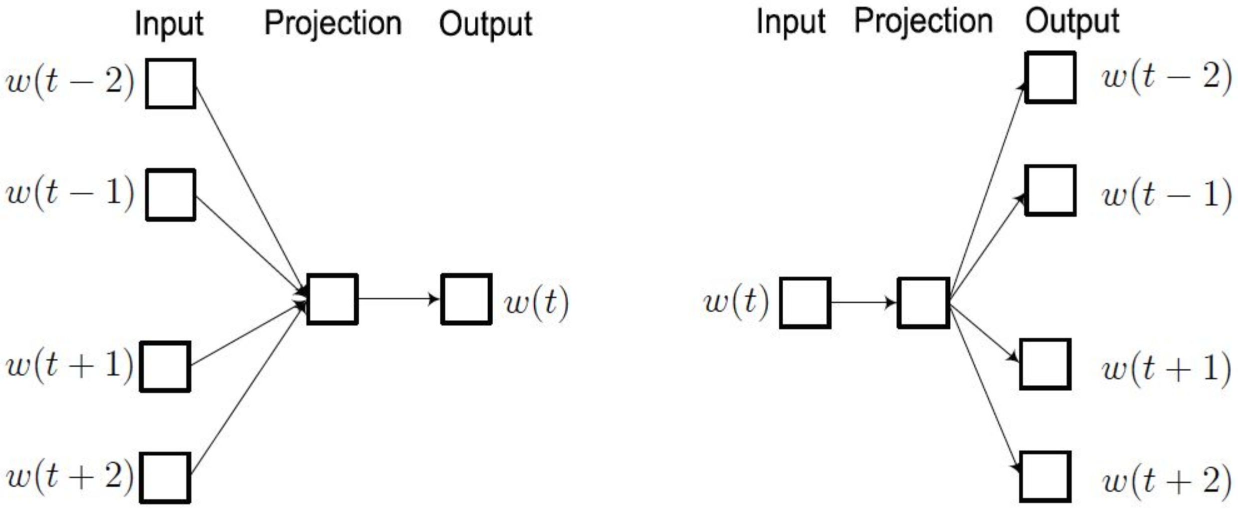
- **Representation Learning**



Representation Learning

Word2Vec

Static Word Embedding



(a) CBOW.　(b) Skip-gram.

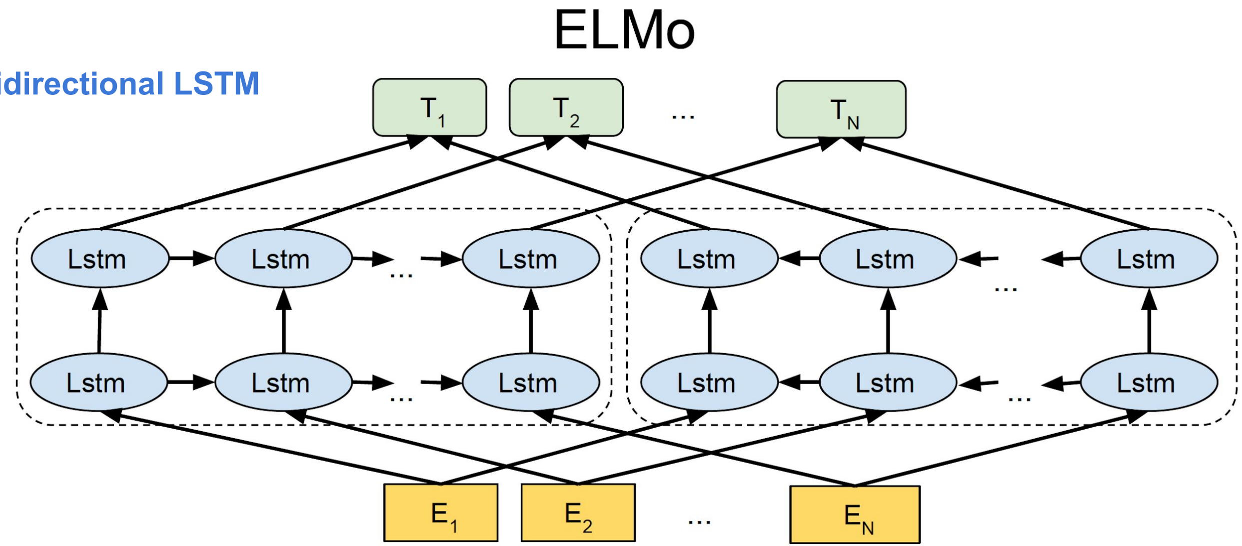
[Mikolov et. al. 2013]

Representation Learning

Word2Vec → ELMO

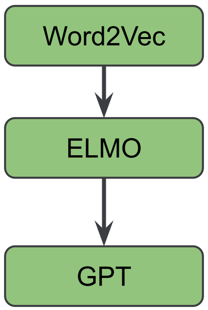
Static Word Embedding

Contextual Word Embedding - Bidirectional LSTM



[Peters et. al. 2018]

Representation Learning

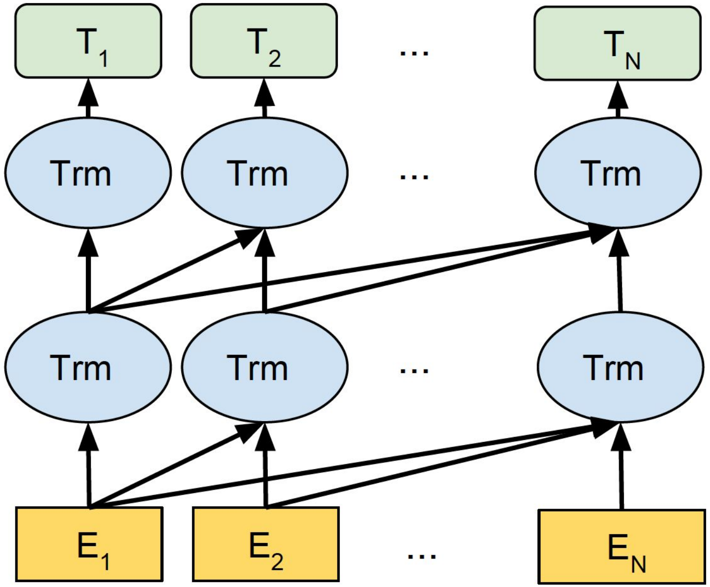


Static Word Embedding

Contextual Word Embedding - Bidirectional LSTM

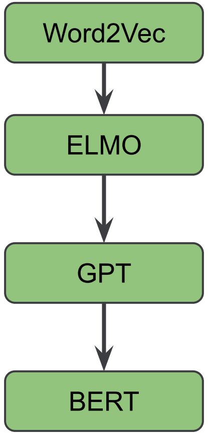
Contextual Word Embedding - One Direction Transformer

OpenAI GPT



[Radford et. al. 2018]

Representation Learning



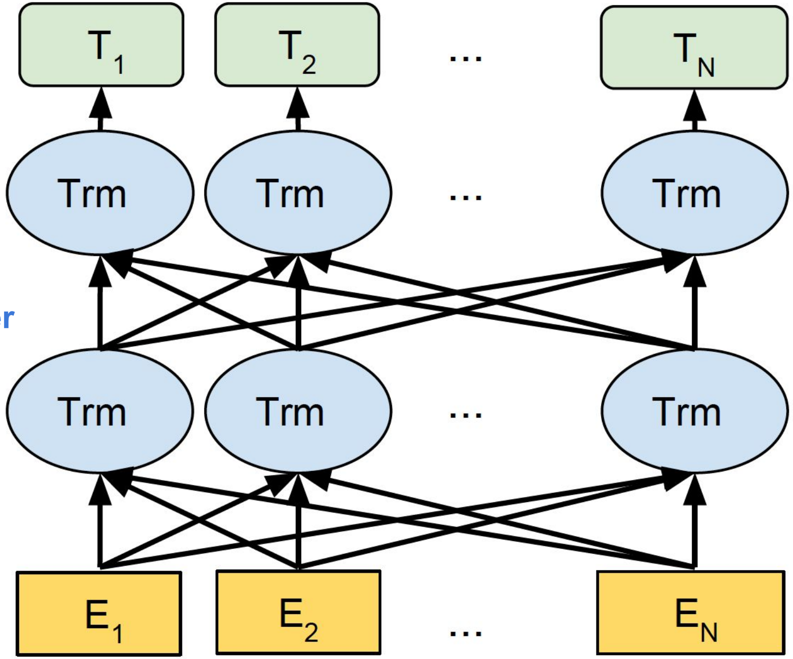
Static Word Embedding

Contextual Word Embedding - **Bidirectional LSTM**

Contextual Word Embedding - **One Direction Transformer**

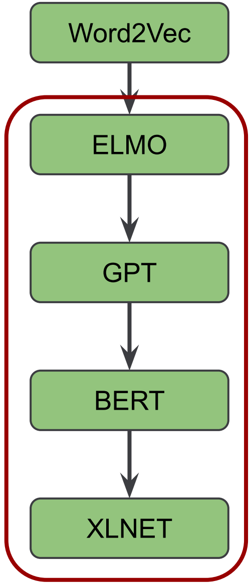
Contextual Word Embedding - **Bidirectional Transformer with Multitask Learning**

BERT



[Devlin et. al. 2018]

Representation Learning



Static Word Embedding

Contextual Word Embedding - Bidirectional LSTM

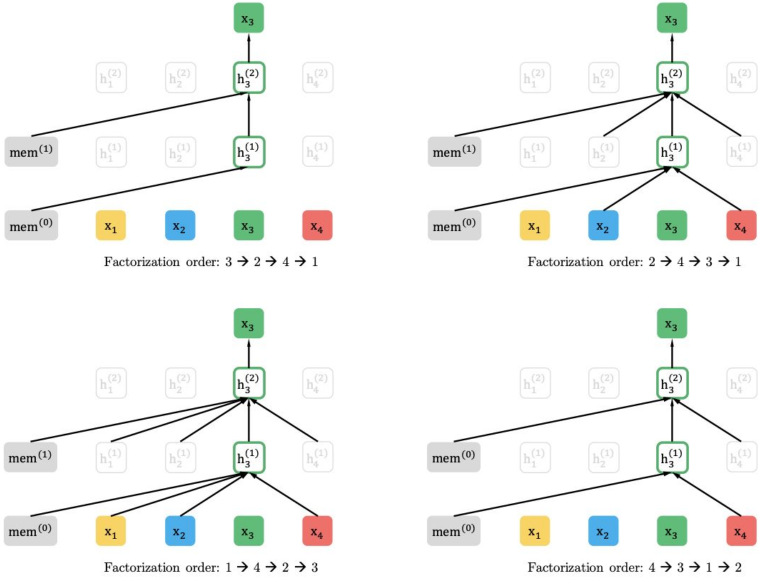
Contextual Word Embedding - One Direction Transformer

Contextual Word Embedding - Bidirectional Transformer with Multitask Learning

Contextual Word Embedding - Bidirectional, Learn correlations among masked tokens

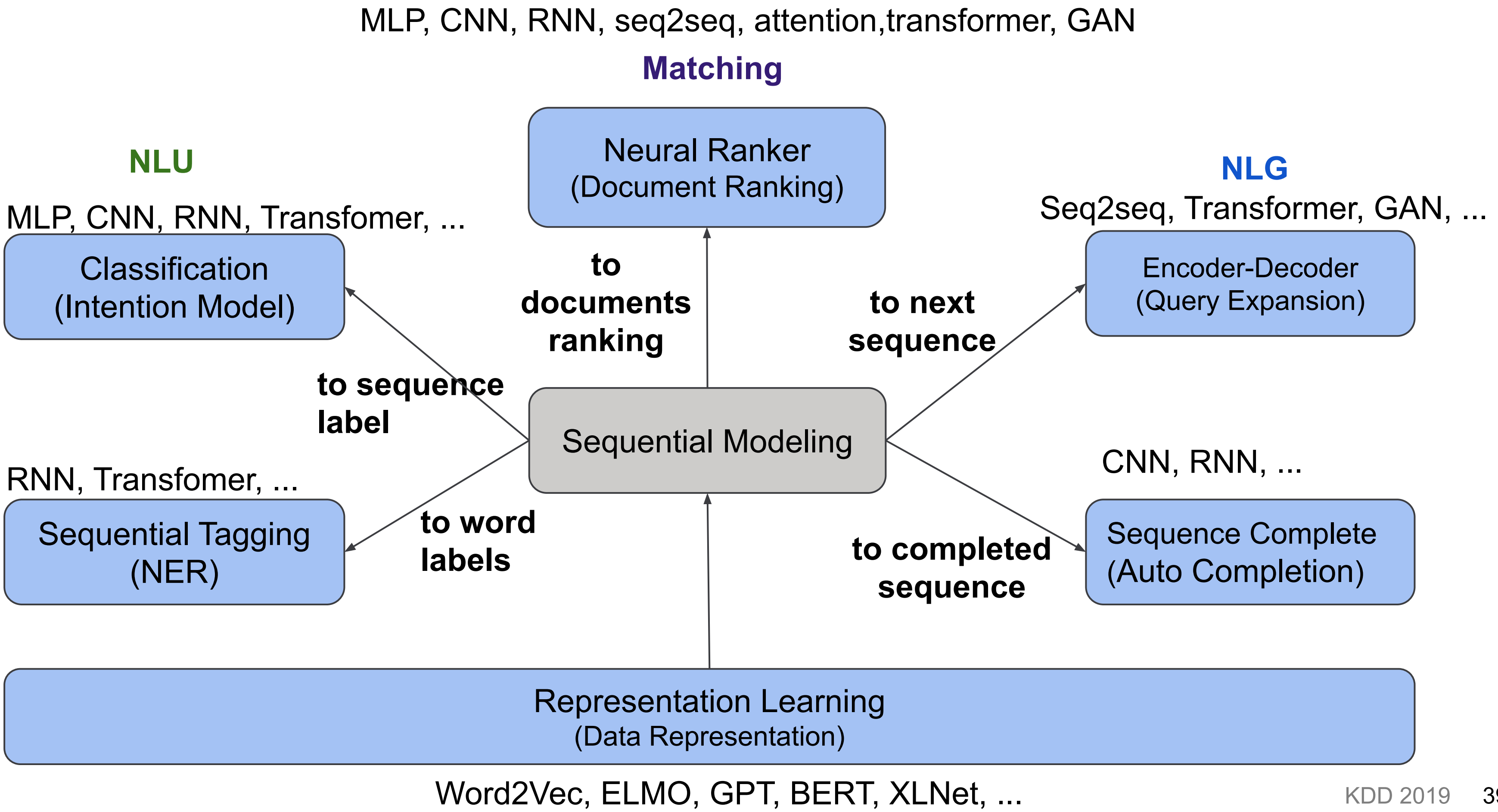
Pre-trained NLP Model

XLNet



[Yang et. al. 2019]

Deep Learning for Natural Language Processing



References - Deep Learning for Natural Language Processing

- [Devlin et. al. 2018] Devlin, Jacob and Chang, Ming-Wei and Lee, Kenton and Toutanova, Kristina. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding, arXiv preprint arXiv:1810.04805, 2018
- [LeCun et. al, 2015] Deep learning, Nature, vol. 521, no. 7553, pp. 436–444, 2015.
- [Lee et. al, 2018] Rare query expansion through generative adversarial networks in search advertising, In Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining, pp. 500-508. ACM, 2018.
- [Mikolov et. al. 2013] Distributed representations of words and phrases and their compositionality, in Advances in neural information processing systems, 2013
- [Mikolov et. al. 2013] Efficient estimation of word representations in vector space, arXiv preprint arXiv:1301.3781, 2013.
- [Peters et. al. 2018] Matthew Peters, Mark Neumann, Mohit Iyyer, Matt Gardner, Christopher Clark, Kenton Lee, and Luke Zettlemoyer. 2018a. Deep contextualized word representations. In NAACL.
- [Radford et.al. 2018] Alec Radford, Karthik Narasimhan, Tim Salimans, and Ilya Sutskever. 2018. Improving language understanding with unsupervised learning. Technical report, OpenAI.
- [Robinson, 2018] https://stackabuse.com/introduction-to-neural-networks-with-scikit-learn, 2018.
- [Shi et. al, 2018] Neural Abstractive Text Summarization with Sequence-to-Sequence Models, arXiv preprint arXiv:1812.02303v2, 2018.
- [Yang et. al. 2019] XLNet: Generalized Autoregressive Pretraining for Language Understanding, arXiv preprint arXiv:1906.08237v1, 2019
- [Young et. al. 2018] Recent Trends in Deep Learning Based Natural Language Processing, arXiv preprint arXiv:1708.02709v8, 2018
- [Zhang et. al. 2015] A sensitivity analysis of (and practitioners' guide to) convolutional neural networks for sentence classification, arXiv preprint arXiv:1510.03820, 2015.

Agenda

1. Introduction
2. Deep Learning for Natural Language Processing
3. **Deep NLP in Search and Recommender Systems**
4. Real World Examples

Deep NLP in Search and Recommender Systems - Language Understanding

Jun Shi

Deep NLP in Search and Recommender Systems

- **Language Understanding**
  - **Entity Tagging: word level prediction**
  - **Entity Disambiguation: knowledge base entity prediction**
  - **Intent Classification: sentence level prediction**
  - **Sentiment Analysis and Opinion Mining**
- Document Retrieval and Ranking
  - Efficient Candidate Retrieval
  - Deep Ranking Models
- Language Generation for Assistance
  - Auto Completion
  - Query Reformulation
  - Spell Correction
  - Conversational Recommendation

Entity Tagging

- Problem statement
- Traditional statistical models
  - Hidden Markov Model
  - Maximum Entropy Markov Model
  - (Semi-Markov) Conditional Random Field
- Deep learning models
  - Input layer
  - Context layer
  - Decoder layer
- Training dataset
- Evaluation

Entity Tagging - Problem Statement

- A named entity, a word or a phrase that clearly identifies one item from a set of other items that have similar attributes. [Li et. al. 2018]
- Entity tagging (Named Entity Recognition, NER), the process of locating and classifying named entities in text into predefined entity categories.

<u>Washington D.C.</u> was named after <u>George Washington</u>. LOCATION PERSON

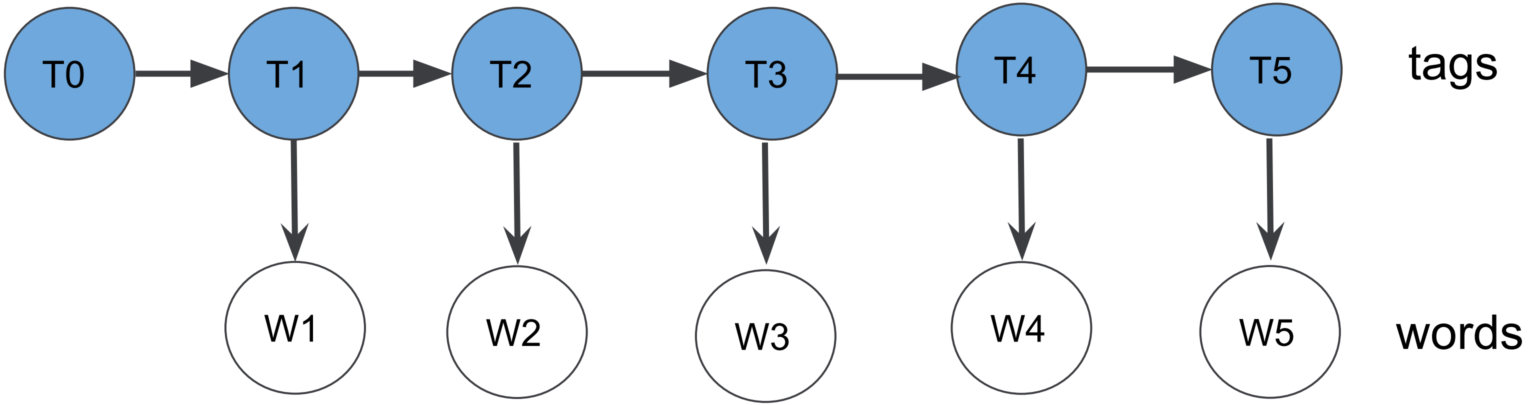
Motivation

- Efficiency
  - Looking only for named entities can speed up search.
- Precision
  - Matching both named entity and tags can increase search precision.
- Quality
  - Ranking retrieved results by considering tags improves search quality.

Traditional Statistical Models

- Generative model
  - Hidden Markov Model [Baum, et. al. 1966]
- Discriminative model
  - Maximum Entropy Markov Model [McCallum, et. al. 2000]
  - (Semi-Markov) Conditional Random Field [Lafferty, et. al. 2001]

Hidden Markov Model

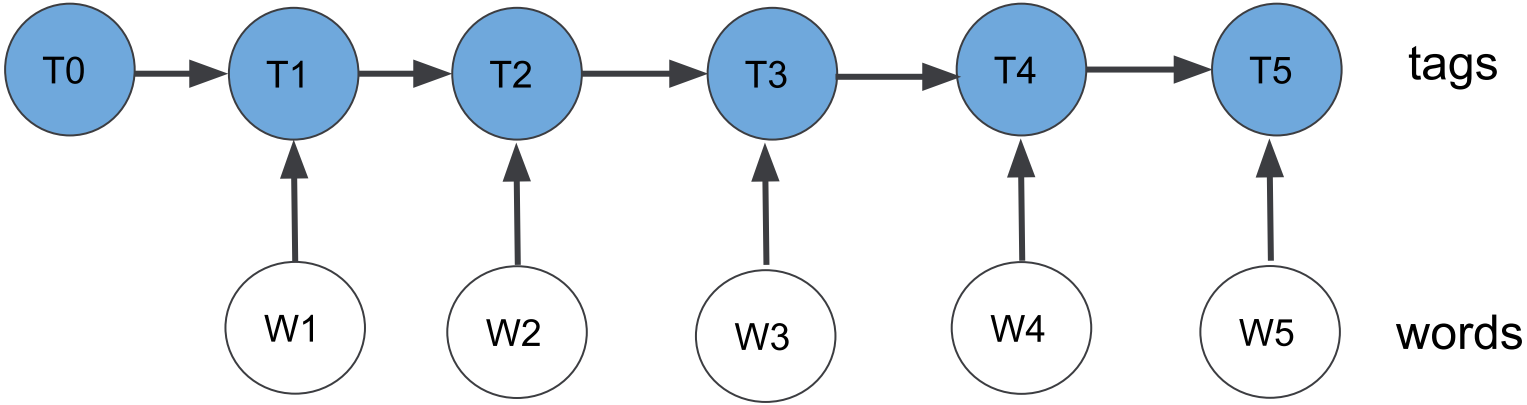


Generative model, model joint probability Pr(T, W) instead of conditional probability Pr(T | W).

$$\Pr(\boldsymbol{T}, \boldsymbol{W}) = \prod_{i=1}^{L} (\Pr(w_i | t_i) \Pr(t_i | t_{i-1}))$$

t_0 is a dummy start state.

Maximum Entropy Markov Model

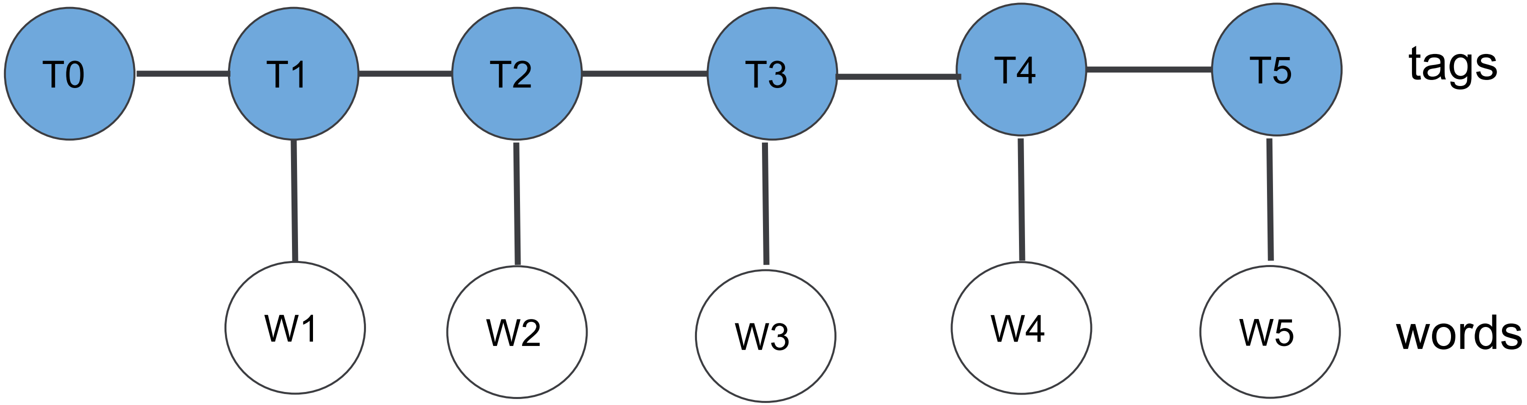


Discriminative model, model conditional probability Pr(T | W) directly.

$$\Pr(\boldsymbol{T}|\boldsymbol{W}) = \prod_{i=1}^{L} \Pr(t_i|t_{i-1}, w_i) = \prod_{i=1}^{L} \frac{\exp(\sum_j \beta_j f_j(t_{i-1}, w_i))}{Z(t_{i-1}, w_i)}$$

t_0 is a dummy start state.

Conditional Random Field

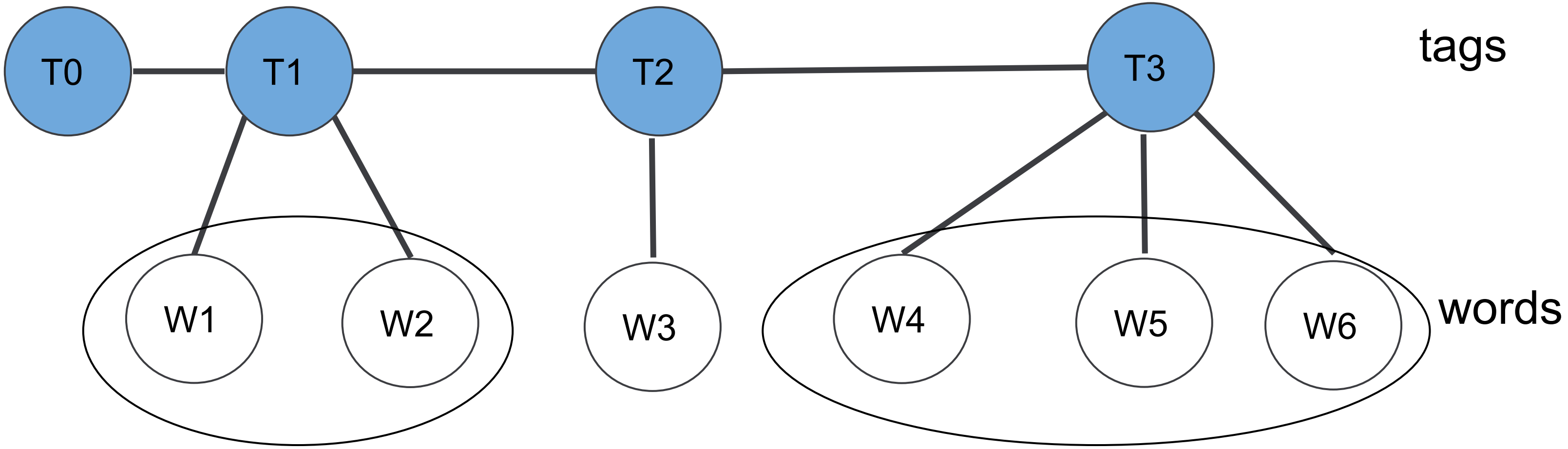


Discriminative model, model conditional probability Pr(T | W) directly.

$$\Pr(\mathbf{T}|\mathbf{W}) = \frac{\prod_{i=1}^{L} \exp(\Sigma_j \beta_j f_j(t_{i-1}, \mathbf{W}))}{Z(\mathbf{T}, \mathbf{W})}$$

t_0 is a dummy start state.

Semi-Markov Conditional Random Field

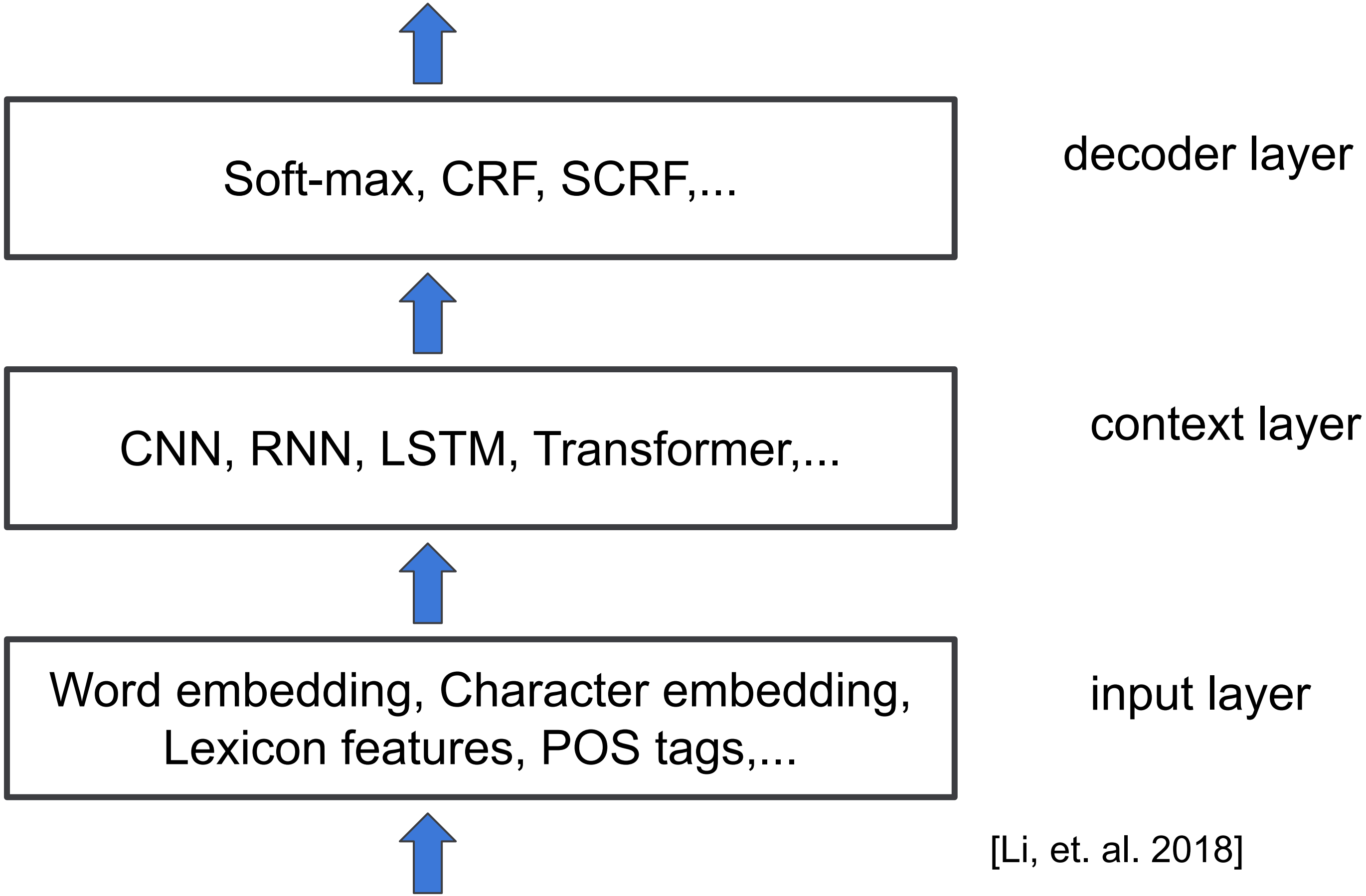


Each tag can correspond to a variable-length phrase.

Summary - Traditional Statistical Models

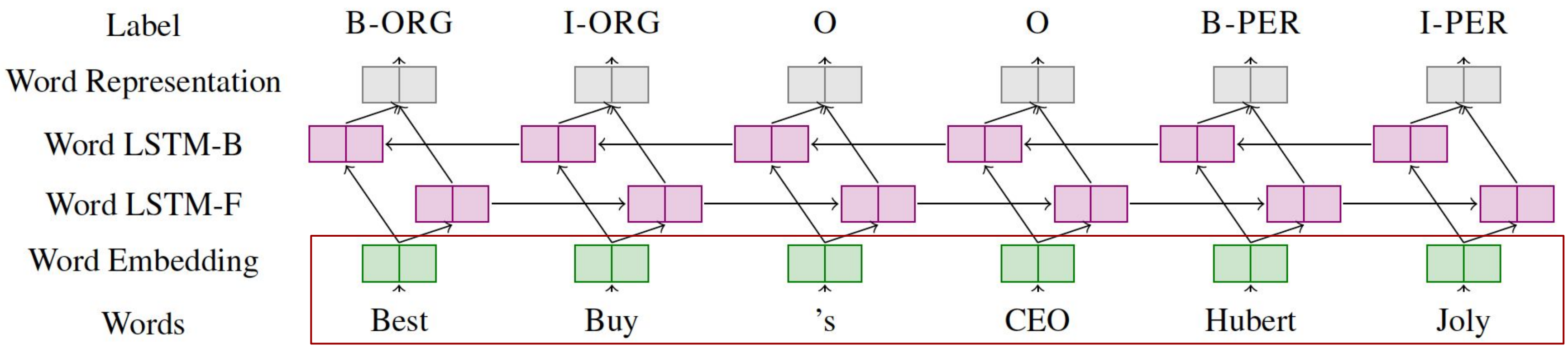
Model	Pros	Cons
Hidden Markov Model	training is simple when states are observed	difficult to include features
Maximum Entropy Markov Model	easy to include features	suffer from “label bias problem” (prefer states with lower number of transitions)
Conditional Random Field	easy to include features, do not suffer from “label bias problem”	training is relatively complex

Deep Learning Tagger Architecture



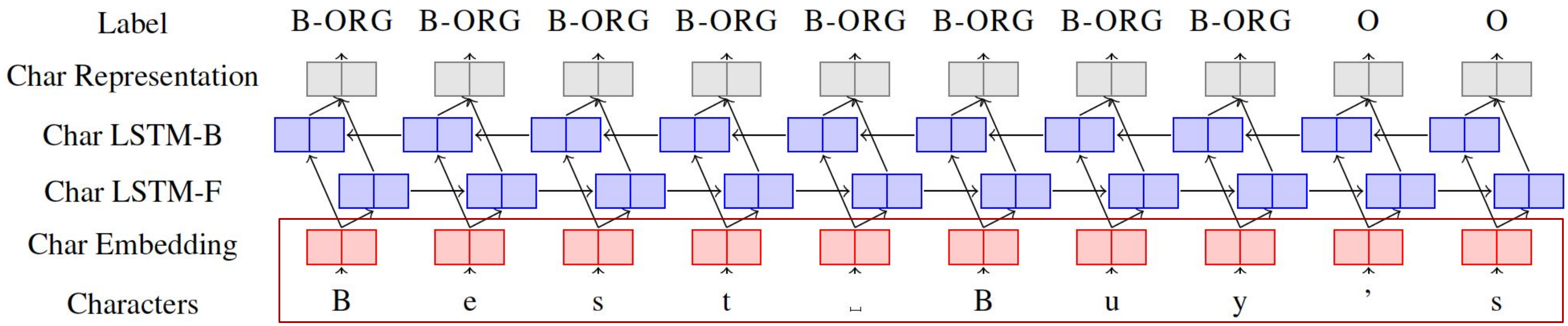
[Li, et. al. 2018]

Input Layer - Word Embedding



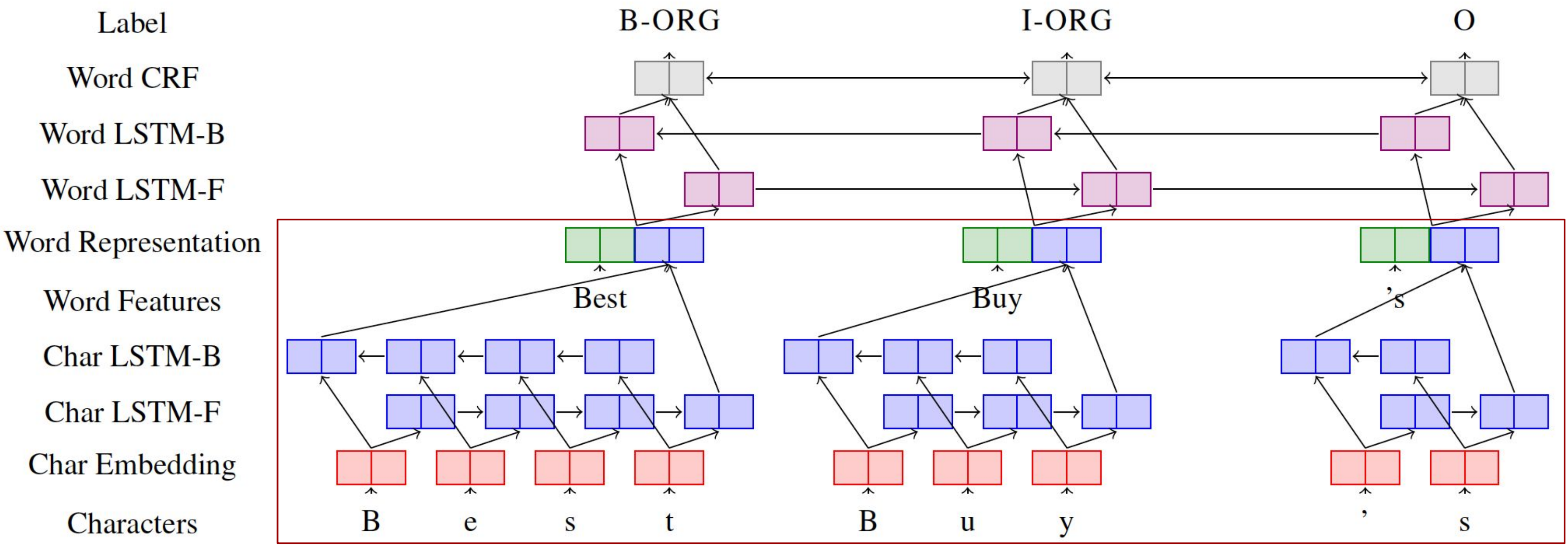
[Yadav, et. al. 2018]

Input Layer - Char Embedding



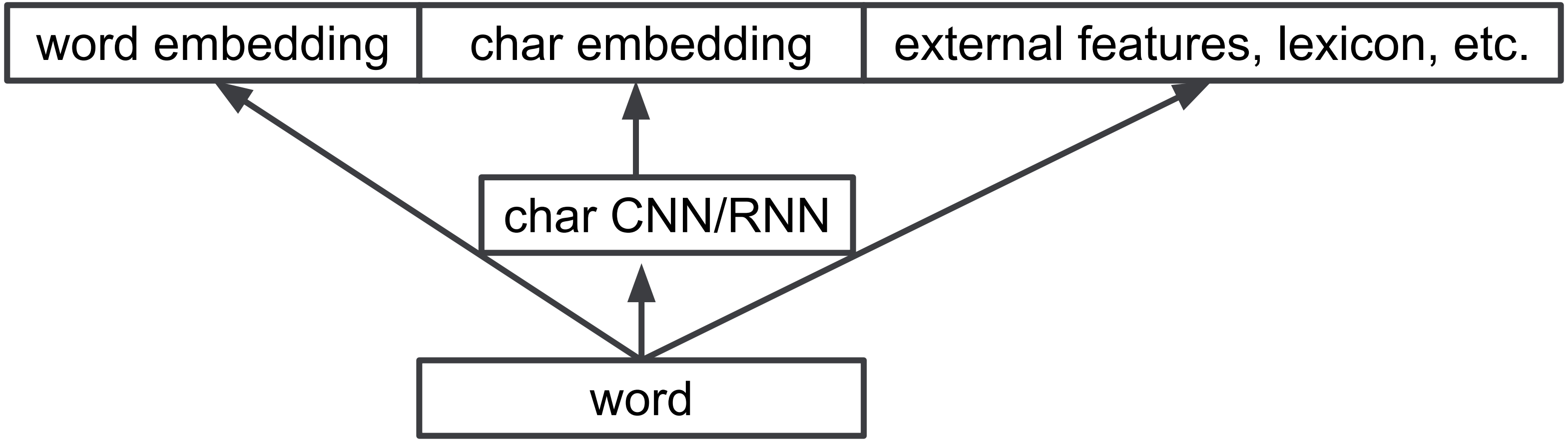
[Yadav, et. al. 2018]

Input Layer - Word and Char Embedding

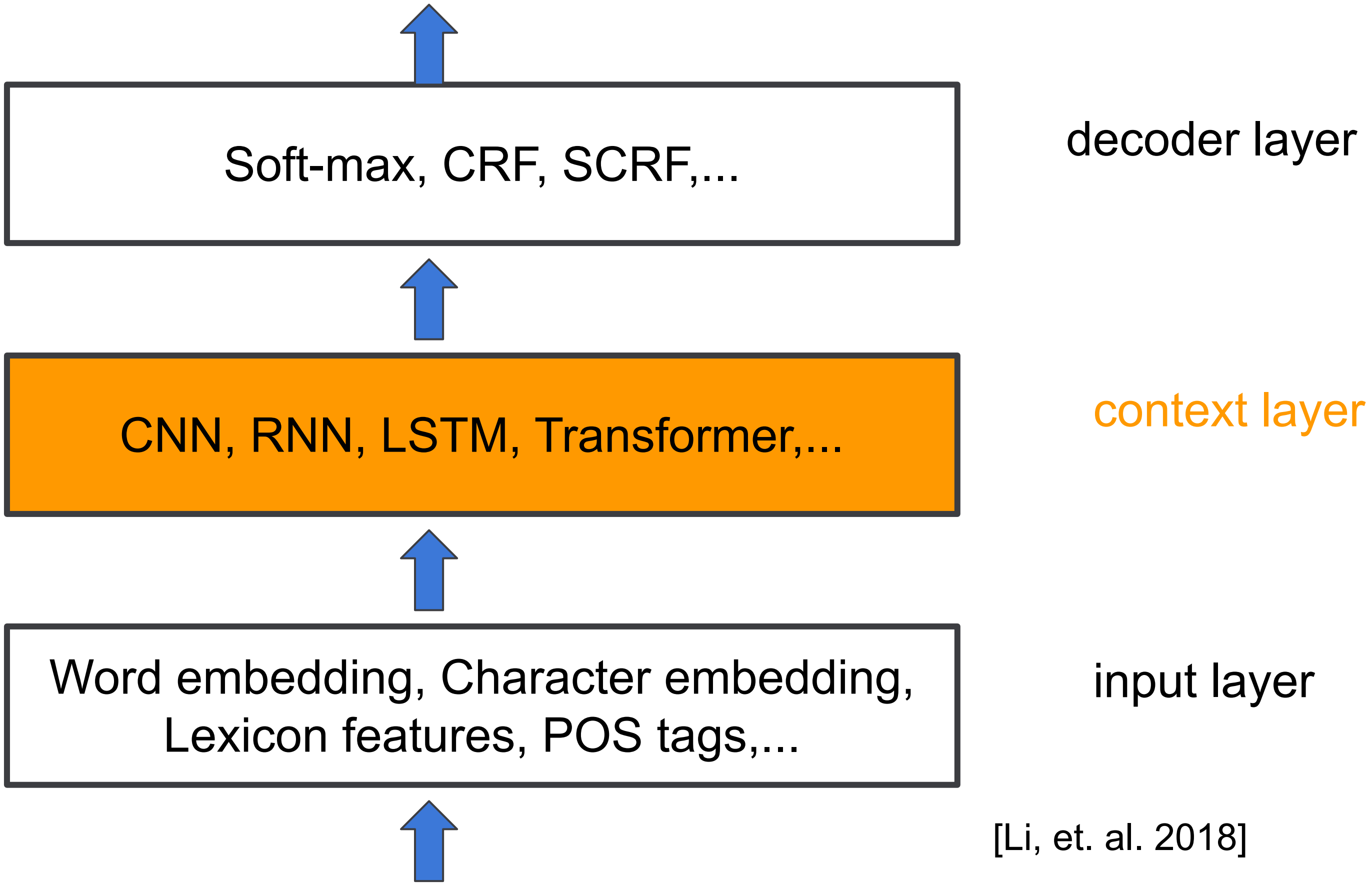


[Yadav, et. al. 2018]

Summary - Input Layer

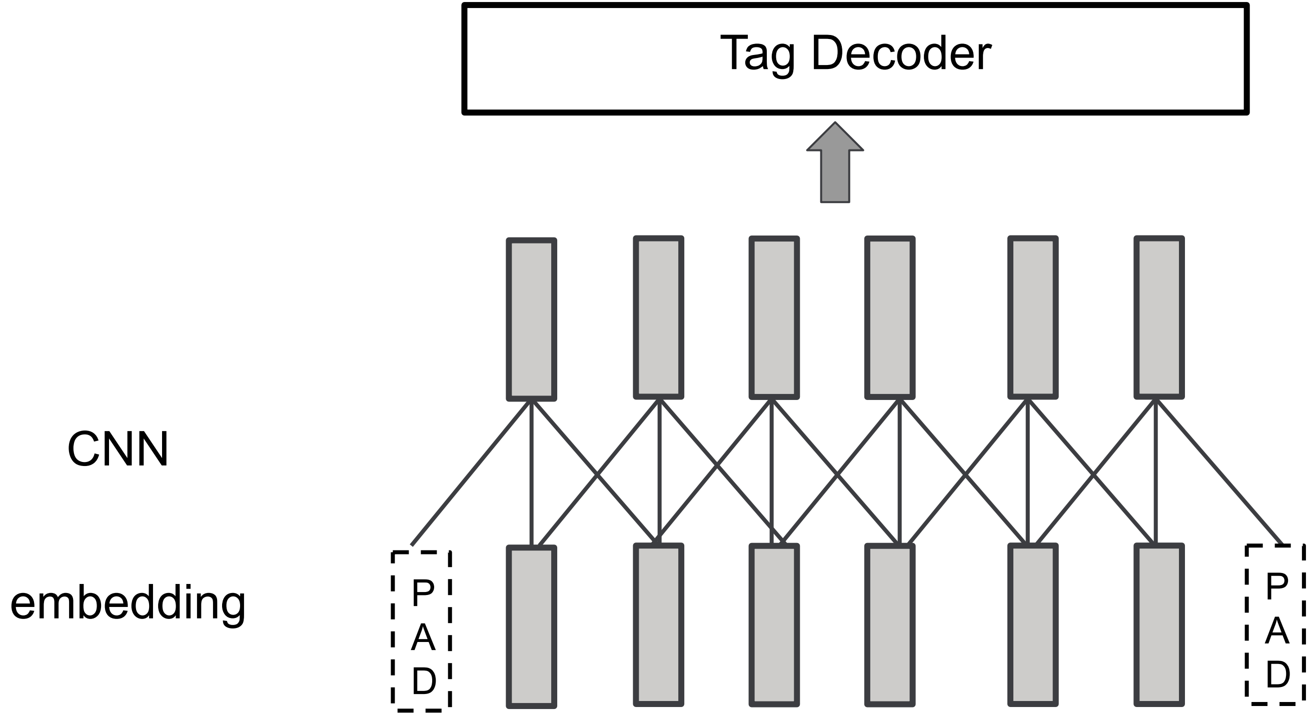


Deep Learning Tagger Architecture



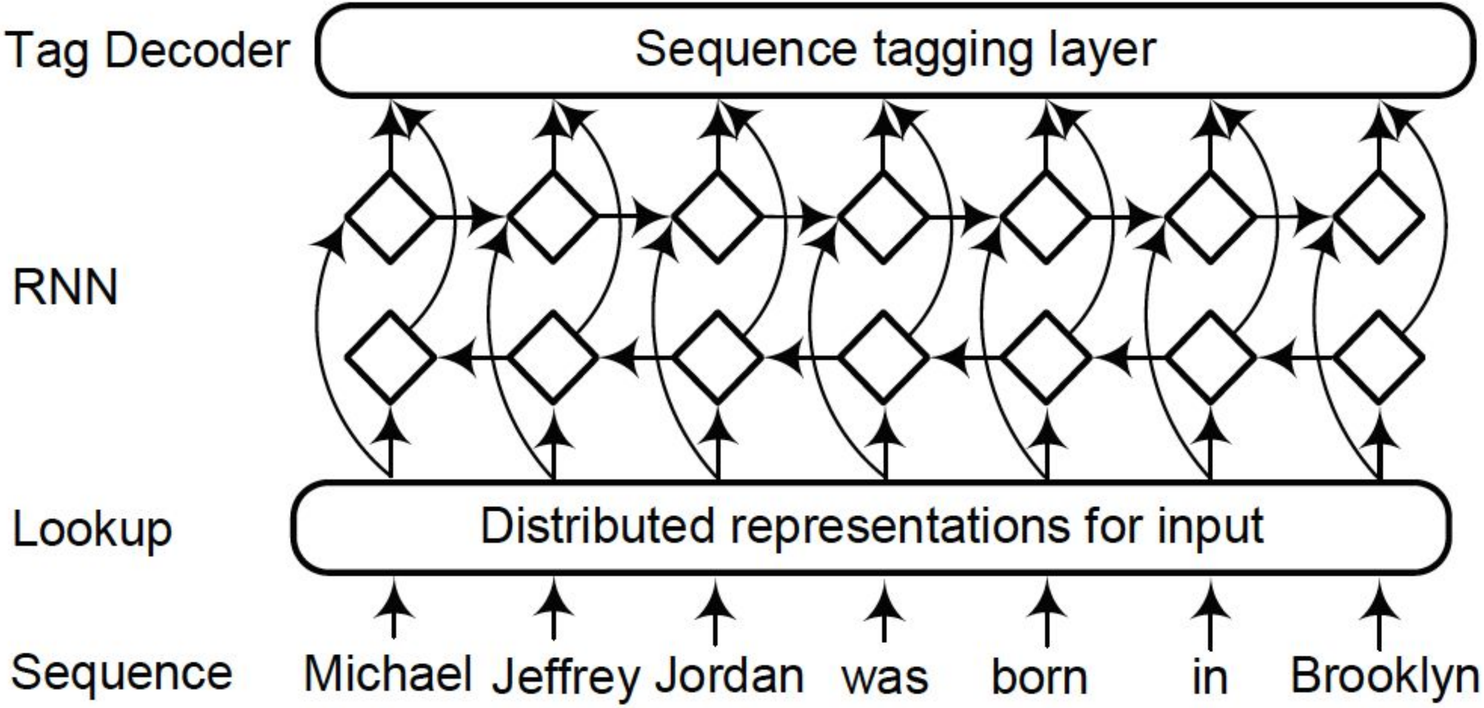
[Li, et. al. 2018]

Context Layer - CNN



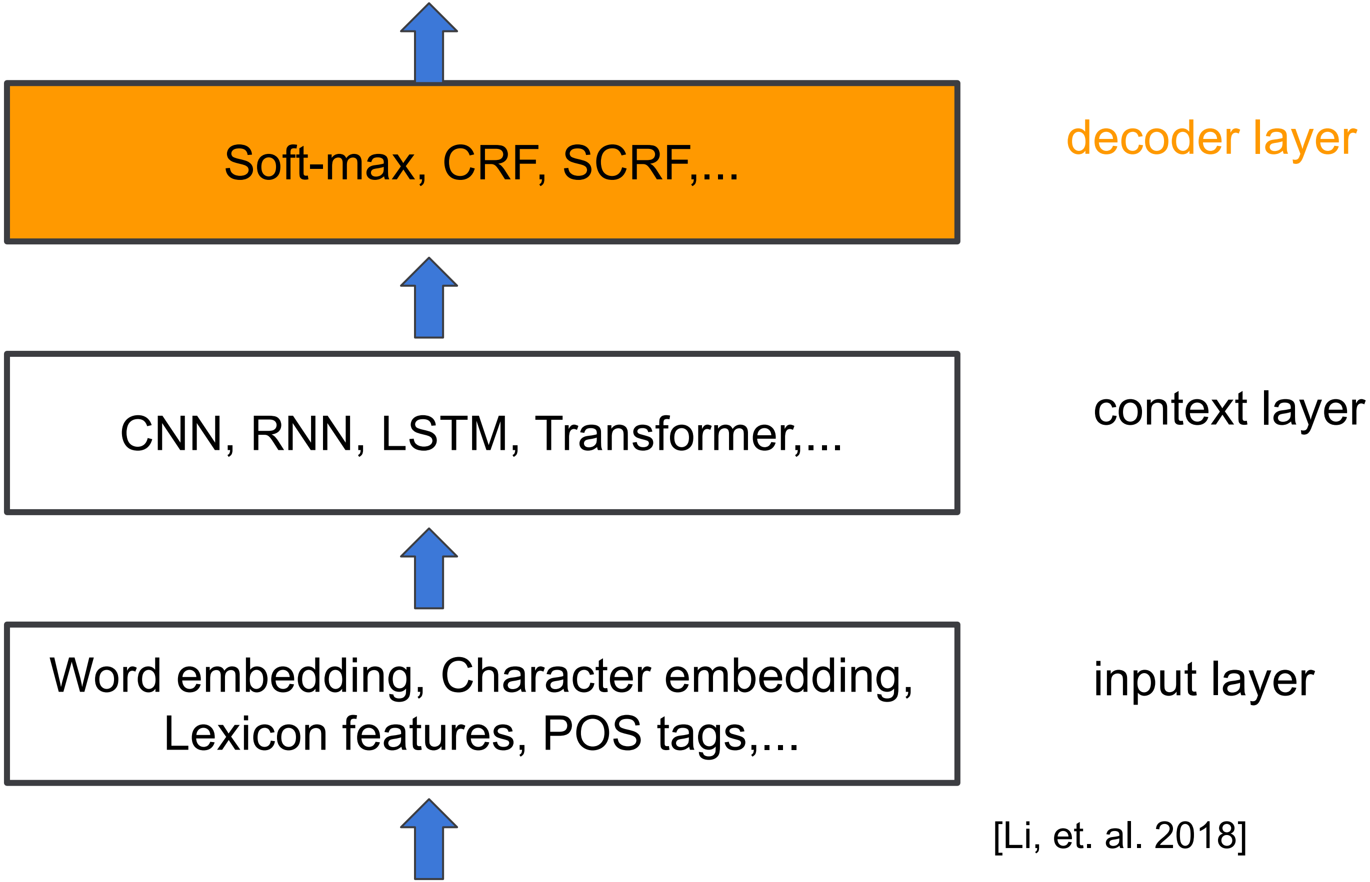
[Li et. al. 2018]

Context Layer - RNN



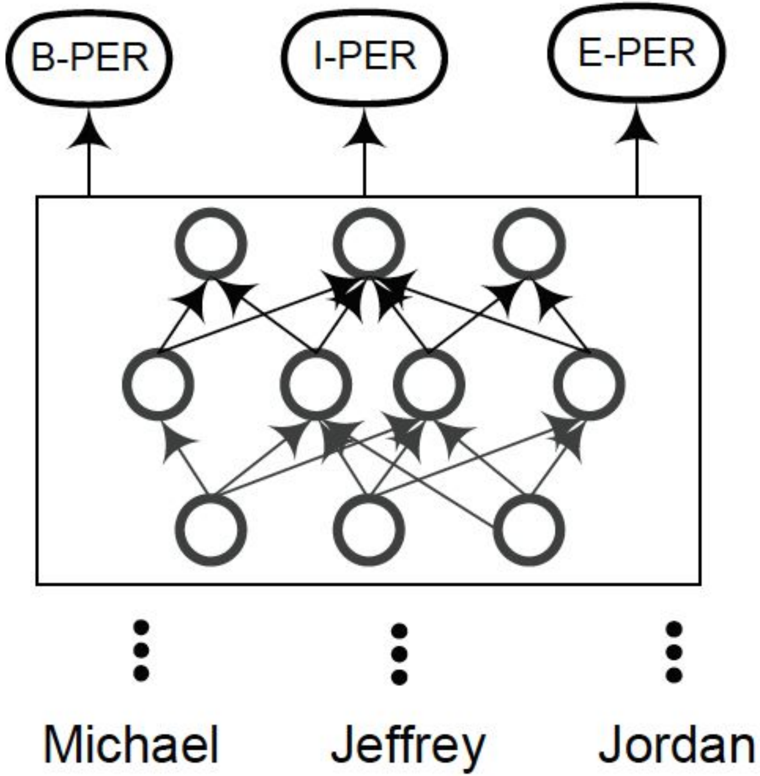
[Li et. al. 2018]

Deep Learning Tagger Architecture



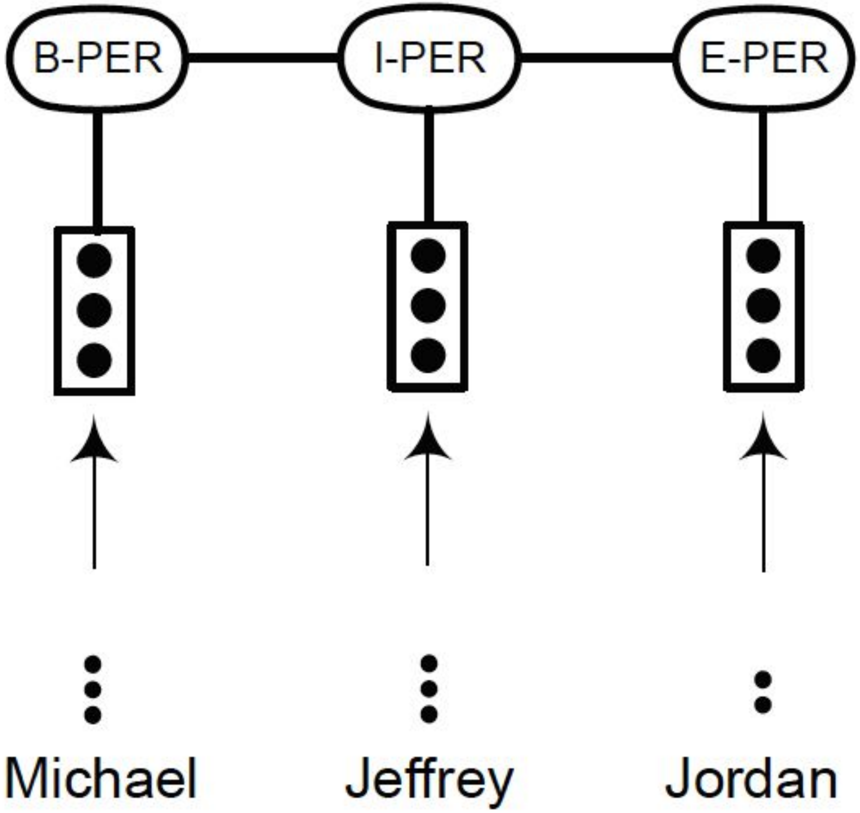
[Li, et. al. 2018]

Tag Decoder - MLP+softmax



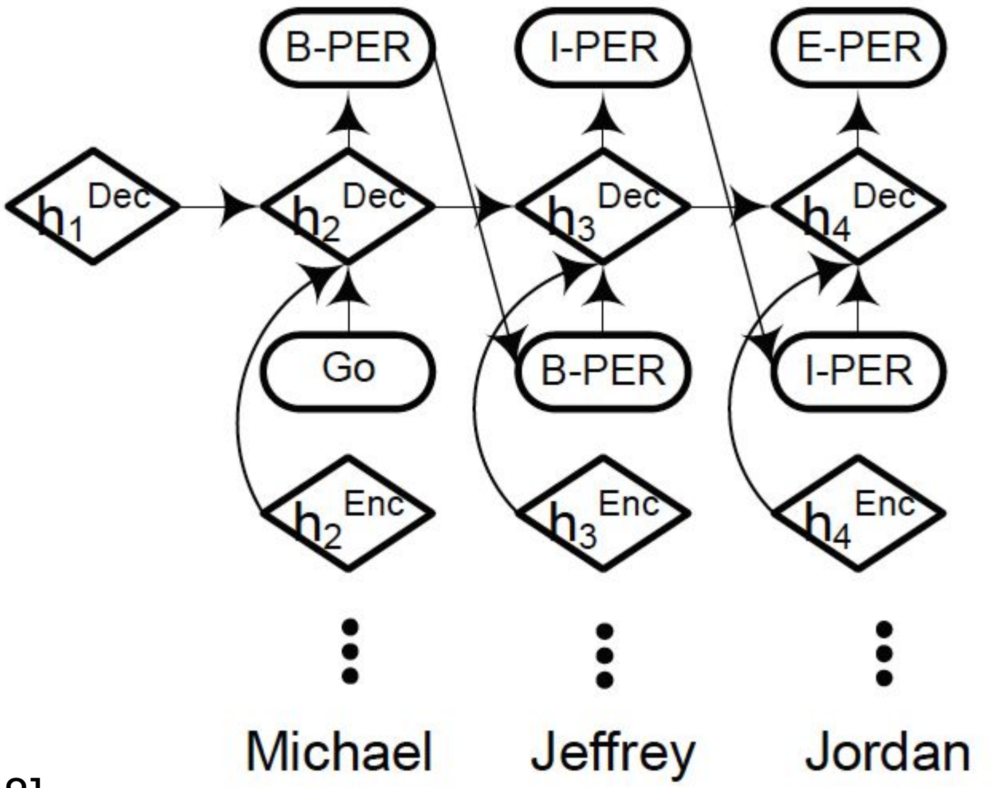
[Li et. al. 2018]

Tag Decoder - CRF



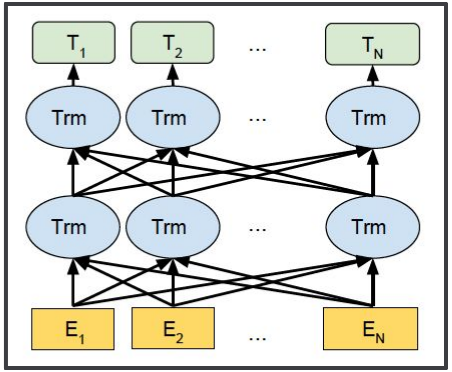
[Li et. al. 2018]

Tag Decoder - RNN



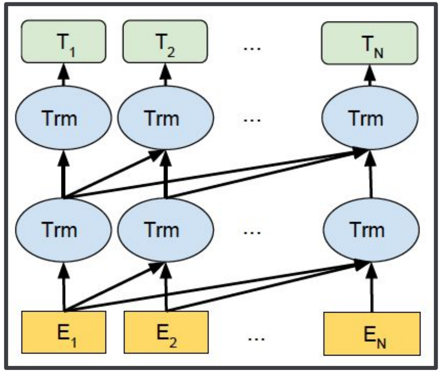
[Li et. al. 2018]

Pre-Training and Fine-Tuning



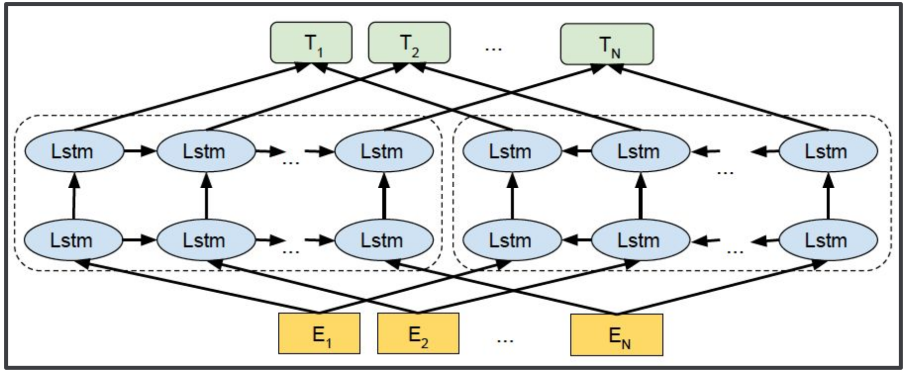
BERT

[Devlin et. al. 2018]



GPT

[Radford et. al. 2018]



ELMo

[Peters et. al. 2018]

Entity Tagging Evaluation

Exact-match evaluation

- segment match
- tag match

$$\text{Precision} = \frac{\text{TP}}{\text{TP} + \text{FP}} \qquad \text{Recall} = \frac{\text{TP}}{\text{TP} + \text{FN}}$$

$$\text{F1-score} = \frac{\text{Precision} * \text{Recall}}{\text{Precision} + \text{Recall}}$$

Entity Tagging Training Dataset

Corpus	Year	Text Source	#Tags	URL
MUC-6	1995	Wall Street Journal texts	7	https://catalog.ldc.upenn.edu/LDC2003T13
MUC-6 Plus	1995	Additional news to MUC-6	7	https://catalog.ldc.upenn.edu/LDC96T10
MUC-7	1997	New York Times news	7	https://catalog.ldc.upenn.edu/LDC2001T02
CoNLL03	2003	Reuters news	4	https://www.clips.uantwerpen.be/conll2003/ner/
ACE	2000 - 2008	Transcripts, news	7	https://www.ldc.upenn.edu/collaborations/past-projects/ace
OntoNotes	2007 - 2012	Magazine, news, conversation, web	89	https://catalog.ldc.upenn.edu/LDC2013T19
W-NUT	2015 - 2018	User-generated text	18	http://noisy-text.github.io
BBN	2005	Wall Street Journal texts	64	https://catalog.ldc.upenn.edu/ldc2005t33
NYT	2008	New York Times texts	5	https://catalog.ldc.upenn.edu/LDC2008T19
WikiGold	2009	Wikipedia	4	https://figshare.com/articles/Learning_multilingual_named_entity_recognition_from_Wikipedia/5462500
WiNER	2012	Wikipedia	4	http://rali.iro.umontreal.ca/rali/en/winer-wikipedia-for-ner
WikiFiger	2012	Wikipedia	113	https://github.com/xiaoling/figer
N^3	2014	News	3	http://aksw.org/Projects/N3NERNEDNIF.html
GENIA	2004	Biology and clinical texts	36	http://www.geniaproject.org/home
GENETAG	2005	MEDLINE	2	https://sourceforge.net/projects/bioc/files/
FSU-PRGE	2010	PubMed and MEDLINE	5	https://julielab.de/Resources/FSU_PRGE.html
NCBI-Disease	2014	PubMed	790	https://www.ncbi.nlm.nih.gov/CBBresearch/Dogan/DISEASE/
BC5CDR	2015	PubMed	3	http://bioc.sourceforge.net/
DFKI	2018	Business news and social media	7	https://dfki-lt-re-group.bitbucket.io/product-corpus/

[Li et. al. 2018]

Entity Tagging on CoNLL03 English

Source	Method	F1 score
[Passos et al. 2014]	CRF	90.90
[Huang et al. 2015]	Bi-LSTM+CRF	84.26
[Collobert et al. 2011]	Conv-CRF	89.59
[Kuru et al. 2016]	char embedding	84.52
[Chiu and Nichols 2015]	word + char embedding	91.62
[Devlin et. al. 2018]	BERT Large	92.8

References - Entity Tagging

- [Baum et. al. 1966] Baum, L. E.; Petrie, T. (1966). "Statistical Inference for Probabilistic Functions of Finite State Markov Chains". The Annals of Mathematical Statistics. 37 (6): 1554–1563.
- [Chiu and Nichols, 2015] Jason PC Chiu and Eric Nichols. 2015. Named entity recognition with bidirectional lstm-cnns. arXiv preprint arXiv:1511.08308
- [Collins and Singer 1999] Michael Collins and Yoram Singer. 1999. Unsupervised models for named entity classification. In 1999 Joint SIGDAT Conference on Empirical Methods in Natural Language Processing and Very Large Corpora.
- [Collobert et. al. 2011] Ronan Collobert, Jason Weston, Leon Bottou, Michael Karlen, Koray Kavukcuoglu, and Pavel Kuksa. 2011. Natural language processing (almost) from scratch. Journal of Machine Learning Research, 12(Aug):2493–2537.
- [Collobert and Weston 2008] Ronan Collobert and Jason Weston. 2008. A unified architecture for natural language processing: Deep neural networks with multitask learning. In Proceedings of the 25th international conference on Machine learning, pages 160–167. ACM.
- [Devlin et. al. 2018] Devlin, Jacob and Chang, Ming-Wei and Lee, Kenton and Toutanova, Kristina. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding, arXiv preprint arXiv:1810.04805, 2018
- [Huang et. al. 2015] Zhiheng Huang, Wei Xu, and Kai Yu. 2015. Bidirectional lstm-crf models for sequence tagging. arXiv preprint arXiv:1508.01991
- [Kuru et. al. 2016] Onur Kuru, Ozan Arkan Can, and Deniz Yuret. 2016. Charner: Character-level named entity recognition. In Proceedings of COLING 2016, the 26th International Conference on Computational Linguistics: Technical Papers, pages 911–921
- [Lafferty et. al. 2001] Lafferty, J., McCallum, A., Pereira, F. (2001). "Conditional random fields: Probabilistic models for segmenting and labeling sequence data". Proc. 18th International Conf. on Machine Learning. Morgan Kaufmann. pp. 282–289
- [Li et. al. 2018] Jing Li, Aixin Sun, Jianglei Han, Chenliang Li, A Survey on Deep Learning for Named Entity Recognition, Dec. 2018, arXiv preprint, https://arxiv.org/abs/1812.09449
- [McCallum et. al. 2000] McCallum, Andrew; Freitag, Dayne; Pereira, Fernando (2000). "Maximum Entropy Markov Models for Information Extraction and Segmentation" (PDF). Proc. ICML 2000. pp. 591–598

References (continued)

- [Passos 2014] Alexandre Passos, Vineet Kumar, and Andrew McCallum. 2014. Lexicon infused phrase embeddings for named entity resolution. arXiv preprint arXiv:1404.5367
- [Peters et. al. 2018] Matthew Peters, Mark Neumann, Mohit Iyyer, Matt Gardner, Christopher Clark, Kenton Lee, and Luke Zettlemoyer. 2018a. Deep contextualized word representations. In NAACL.
- [Radford et.al. 2018] Alec Radford, Karthik Narasimhan, Tim Salimans, and Ilya Sutskever. 2018. Improving language understanding with unsupervised learning. Technical report, OpenAI.
- [Yadav et. al. 2018] Vikas Yadav, Steven Bethard, A Survey on Recent Advances in Named Entity Recognition from Deep Learning models, Proceedings of the 27th International Conference on Computational Linguistics, pages 2145–2158 Santa Fe, New Mexico, USA, August 20-26, 2018.

Deep NLP in Search and Recommender Systems

- Language Understanding
  - Entity Tagging: word level prediction
  - **Entity Disambiguation: knowledge base entity prediction**
  - Intent Classification: sentence level prediction
  - Sentiment Analysis and Opinion Mining
- Document Retrieval and Ranking
  - Efficient Candidate Retrieval
  - Deep Ranking Models
- Language Generation for Assistance
  - Auto Completion
  - Query Suggestion
  - Spell Correction
  - Conversational Recommendation

Entity Disambiguation

- Problem statement
- Motivation
- Challenges
- System architecture
  - Candidate Entity Generation
  - Candidate Entity Selection
  - Joint entity tagging and disambiguation
- Evaluation

Entity Disambiguation - Problem Statement

- Also known as entity linking
- Resolves mentions to entities in a given knowledge base
  - Mentions are mostly named entities
  - Example of knowledge base: freebase, wikipedia
- Examples: Jaguar
  - The prey saw the <u>jaguar</u> cross the jungle. *KB:Jaguar*
  - The man saw a <u>jaguar</u> speed on the highway. *KB:Jaguar_Cars*

Motivation

Increase the quality of the retrieved results

Example:

Which Michael Jordan were you looking for?

Michael Jordan (disambiguation)

From Wikipedia, the free encyclopedia

Michael or **Mike Jordan** may refer to:

People [edit]

Sports [edit]

- Michael Jordan (born 1963), American basketball player and businessman
- Michael Jordan (footballer) (born 1986), English goalkeeper
- Mike Jordan (racing driver) (born 1958), English racing driver
- Mike Jordan (baseball, born 1863) (1863–1940), baseball player
- Mike Jordan (cornerback) (born 1992), American football cornerback
- Michael Jordan (offensive lineman), American football offensive lineman
- Michael-Hakim Jordan (born 1977), American professional basketball player
- Michal Jordán (born 1990), Czech ice hockey player

Other people [edit]

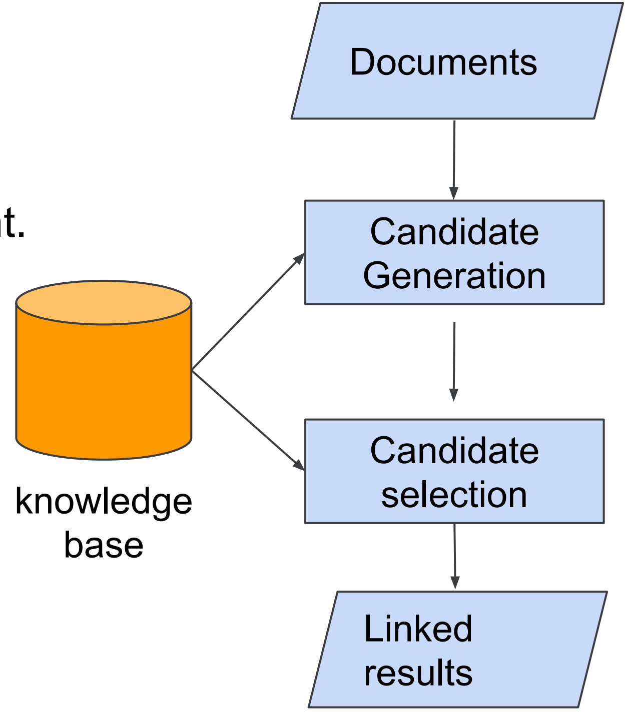
- Michael B. Jordan (born 1987), American actor
- Michael I. Jordan (born 1956), American researcher in machine learning and artificial intelligence
- Michael Jordan (insolvency baron) (born 1931), English businessman
- Michael Jordan (Irish politician), Irish Farmers' Party TD from Wexford, 1927–1932
- Michael H. Jordan (1936–2010), American executive for CBS, PepsiCo, Westinghouse
- Michael Jordan (mycologist), English mycologist

Challenges

- Name variation: *New York* vs *Big Apple*
- Ambiguity: *Michael Jordan*
- Metonymy: *Beijing* (city or Chinese government)
- Absence: no entries in knowledge base
- Evolving information: new company names, e.g. *tic-tok*.

System Architecture

1. Candidate entity generation
    a. Name dictionary based techniques.
    b. Surface form expansion from the local document.
    c. Methods based on search engines.
2. Candidate entity selection
    a. Graph-based methods
    b. Text-based methods



Candidate Entity Generation

- Name dictionary based techniques
  - Entity pages
  - Redirect pages
  - Disambiguation pages
  - Bold phrases from the first paragraphs
  - Hyperlinks in Wikipedia articles
- Surface form expansion
  - Heuristic based methods
  - Supervised learning methods
- Methods based on search engines

[Shen et. al. 2014]

Michael Jordan (disambiguation)

From Wikipedia, the free encyclopedia

Michael or **Mike Jordan** may refer to:

People [edit]

Sports [edit]

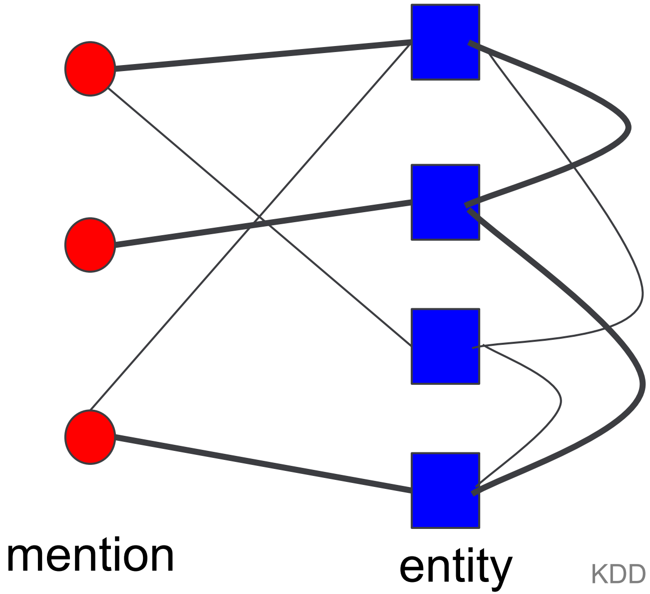
- Michael Jordan (born 1963), American basketball player and businessman
- Michael Jordan (footballer) (born 1986), English goalkeeper
- Mike Jordan (racing driver) (born 1958), English racing driver
- Mike Jordan (baseball, born 1863) (1863–1940), baseball player
- Mike Jordan (cornerback) (born 1992), American football cornerback
- Michael Jordan (offensive lineman), American football offensive lineman
- Michael-Hakim Jordan (born 1977), American professional basketball player
- Michal Jordán (born 1990), Czech ice hockey player

Other people [edit]

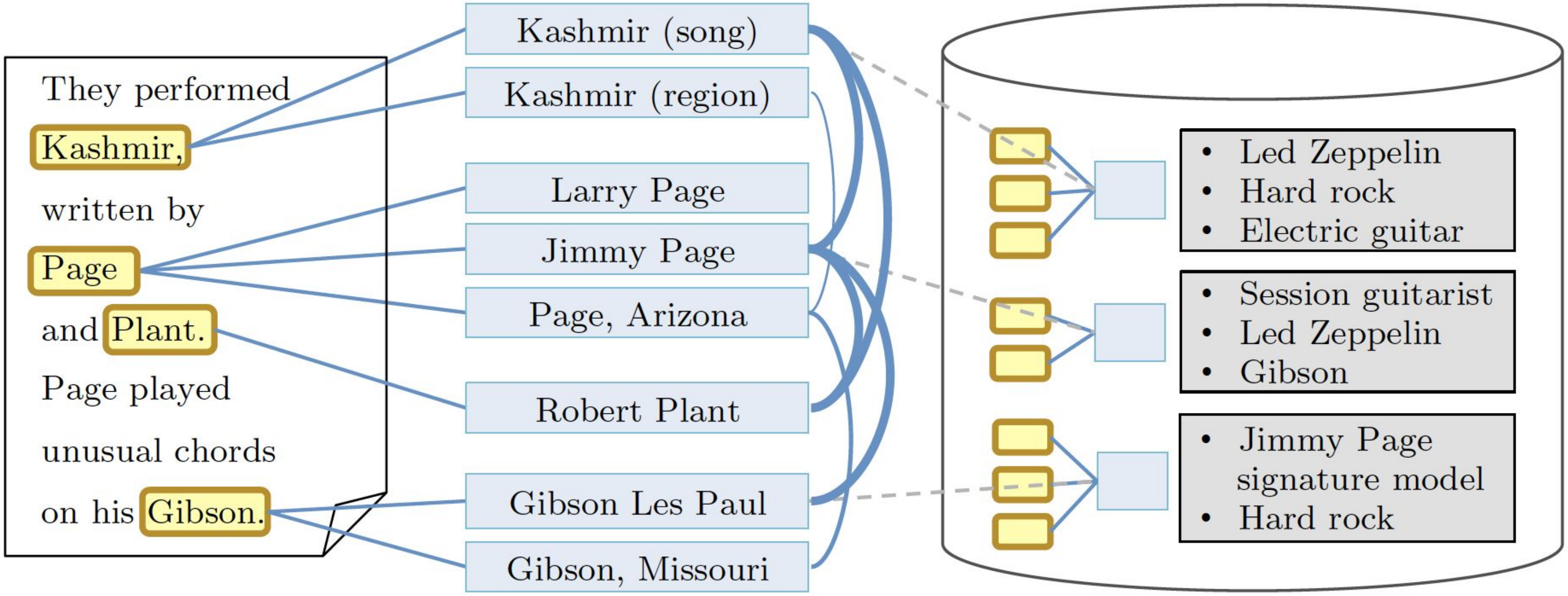
- Michael B. Jordan (born 1987), American actor
- Michael I. Jordan (born 1956), American researcher in machine learning and artificial intelligence
- Michael Jordan (insolvency baron) (born 1931), English businessman
- Michael Jordan (Irish politician), Irish Farmers' Party TD from Wexford, 1927–1932
- Michael H. Jordan (1936–2010), American executive for CBS, PepsiCo, Westinghouse
- Michael Jordan (mycologist), English mycologist

Candidate Entity Selection - Graph based models

- knowledge graph lends itself to graph based methods.
- Build a weighted undirected graph with mentions and candidate entities as nodes
  - weights on mention-entity edges.
  - weights on entity-entity edge.
- Compute an optimal subgraph:
  - optimality is model dependent.
  - contains all mentions
  - one entity per mention



An Example Graph



[Hoffart, et. al. 2011]

Candidate Entity Selection - Text-Based Models

- Idea: find agreement (similarity) between entity and mention.
  - Example: computational intelligence by *Michael Jordan* and Stuart Russell from UC Berkeley

- wiki/Michael_Jordan
- wiki/Michael_I._Jordan

- Models according to context.
  - Direct Models.
    - use mention and candidate entities only.
  - Local Models
    - use local context around the mention.
  - Coherence Models.
    - use mentions and entities in a document.
  - Collaborative Models.
    - use mentions and entities across related documents.

increasing context

Individual vs Joint Approach

Two types of approaches according to how mentions are resolved.

- Individual approach
  - Handle one mention at a time.
  - Rank the candidate entities.
  - Relatively low complexity.
- Joint approach
  - Treat all mentions in a document as a sequence.
  - Jointly optimize the mention-entity pairs.
  - High complexity, usually resorting to approximations.

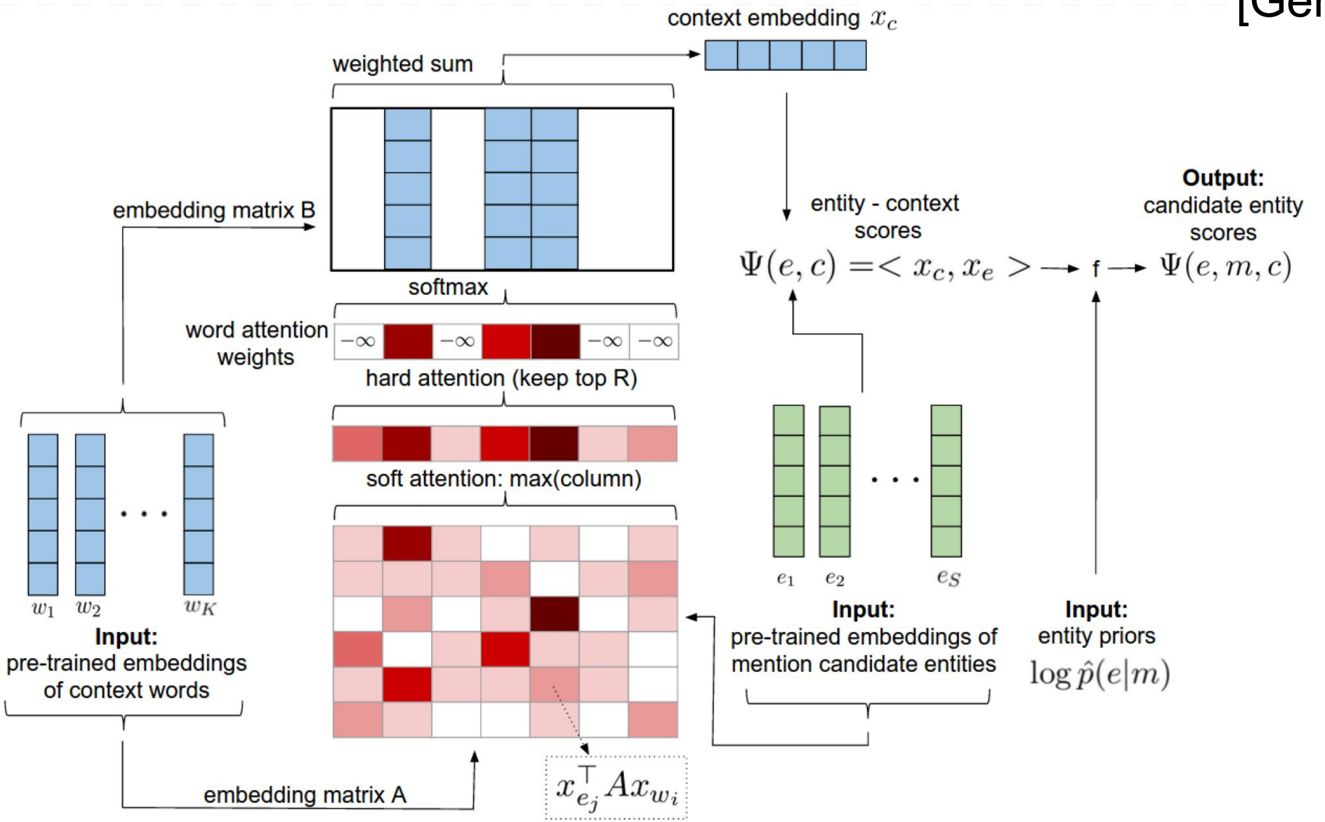
Features

Different features are used in candidate selection modeling.

- Traditional Features
  - entity popularity, entity type.
  - surface form similarity between entity and mention.
  - coherence between mapped entities
- Deep Features [Yamada et. al. 2016]
  - word embedding
  - entity embedding
  - similarity between words, context and candidate entities.

Example: Entity Disambiguation with Attention

[Genea et. al. 2017]



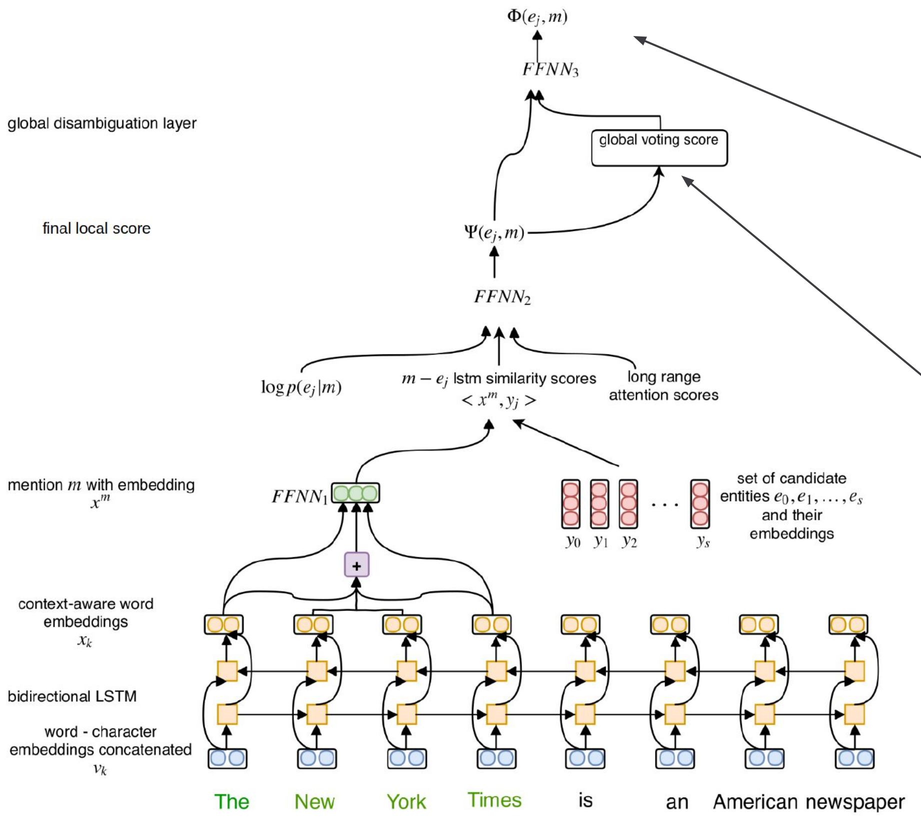
training:
max-margin loss

Joint Entity Tagging and Disambiguation

- Previously, we find entities first, then link them. But if entities are identified wrong, disambiguation will likely fail too.
  - Over-split: Romeo and Juliet by Shakespeare
  - Under-split: Baby Romeo and Juliet were born hours apart.
- We could do entity tagging and disambiguation jointly. [Sil, et. al.2013]
  - Over-generate candidate mentions
  - Generate possible entities per mention
  - Score non-overlapping mention-entity pair jointly.

A Joint Neural Entity Tagging and Disambiguation

[Kolitsas, et. al. 2018]



- Trained on golden mention-entity pairs.
- Used max-margin loss.

cosine similarity between current entity and average of other entities in the document

Select mentions that have at least one possible entity

Entity Disambiguation Evaluation

- Entity-tagging-style F1 score
  - A link is considered correct only if the mention matches the gold boundary and the linked entity is also correct
- Accuracy
  - another common metric, simpler than F1 score.
- TAC-KBP B-Cubed+ F1 score
  - not widely used

References - Named Entity Disambiguation

- [Cucerzan 2007] Cucerzan, Silviu. Large-Scale Named Entity Disambiguation Based on Wikipedia Data. Proceedings of the 2007 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning (EMNLP-CoNLL): 708–716.
- [Ganea et. al. 2017] Octavian-Eugen Ganea and Thomas Hofmann. 2017. Deep joint entity disambiguation with local neural attention. In Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing, pages 2609–2619. Association for Computational Linguistics.
- [Globerson et. al. 2016] Amir Globerson, Nevena Lazic, Soumen Chakrabarti, Amarnag Subramanya, Michael Ringgaard, and Fernando Pereira. 2016. Collective entity resolution with multi-focal attention. In ACL (1).
- [Han et. al. 2011] Han, Xianpei; Sun, Le; Zhao, Jun. Collective Entity Linking in Web Text: A Graph-based Method.
- [Hoffart, et. al. 2011] Johannes Hoffart, Mohamed A. Yosef, Ilaria Bordino, Hagen F¨urstenau, Manfred Pinkal, Marc Spaniol, Bilyana Taneva, Stefan Thater, and Gerhard Weikum. 2011. Robust disambiguation of named entities in text. In Proceedings of the Conference on Empirical Method in Natural Language Processing, pages 782–792. Association for Computational Linguistics.
- [Kolitsas et al., 2018] Nikolaos Kolitsas, Octavianeugen Ganea, and Thomas Hofmann. End-to-end neural entity linking. In CoNLL, 2018.
- [Kulkarni et. al. 2009] S. Kulkarni, A. Singh, G. Ramakrishnan, and S. Chakrabarti, Collective annotation of Wikipedia entities in web text, in SIGKDD, 2009, pp. 457–466
- [Ling et al.2015] Xiao Ling, Sameer Singh, and Daniel S. Weld. 2015. Design Challenges for Entity Linking. Transactions of the Association for Computational Linguistics, 3:315–328.
- [Le et. al., 2018] Phong Le and Ivan Titov. Improving entity linking by modeling latent relations between mentions. In ACL, 2018.
- [Rao et. al. 2013] Rao, Delip; McNamee, Paul; Dredze, Mark (2013). Entity Linking: Finding Extracted Entities in a Knowledge Base. Multi-source, Multilingual Information Extraction and Summarization. Springer Berlin Heidelberg: 93–115

References (continued)

- [Raiman et. al. 2018] Jonathan Raiman and Olivier Raiman. 2018. DeepType: Multilingual Entity Linking by Neural Type System Evolution. In Proc. of AAAI.
- [Shen et. al. 2014] Wei Shen, Jianyong Wang, and Jiawei Han. 2014. Entity linking with a knowledge base: Issues, techniques, and solutions. TKDE
- [Sil et. al. 2013] Avirup Sil and Alexander Yates. 2013. Re-ranking for joint named-entity recognition and linking. In Proceedings of the 22nd ACM international conference on Conference on information & knowledge management, pages 2369–2374. ACM
- [Sil et. al. 2016] Sil, A., and Florian, R. 2016. One for all: Towards language independent named entity linking. ACL.
- [Yamada et. al. 2016] Ikuya Yamada, Hiroyuki Shindo, Hideaki Takeda, and Yoshiyasu Takefuji. 2016. Joint learning of the embedding of words and entities for named entity disambiguation. CoNLL 2016, page 250.

Deep NLP in Search and Recommender Systems

- Language Understanding
  - Entity Tagging: word level prediction
  - Entity Disambiguation: knowledge base entity prediction
  - **Intent Classification: sentence level prediction**
  - Sentiment Analysis and Opinion Mining
- Document Retrieval and Ranking
  - Efficient Candidate Retrieval
  - Deep Ranking Models
- Language Generation for Assistance
  - Auto Completion
  - Query Reformulation
  - Spell Correction
  - Conversational Recommendation

Intent Classification

- Problem statement
- Deep Learning Models
  - fastText
  - CNN
  - Bi-RNN + Attention

Intent Classification - Problem Statement

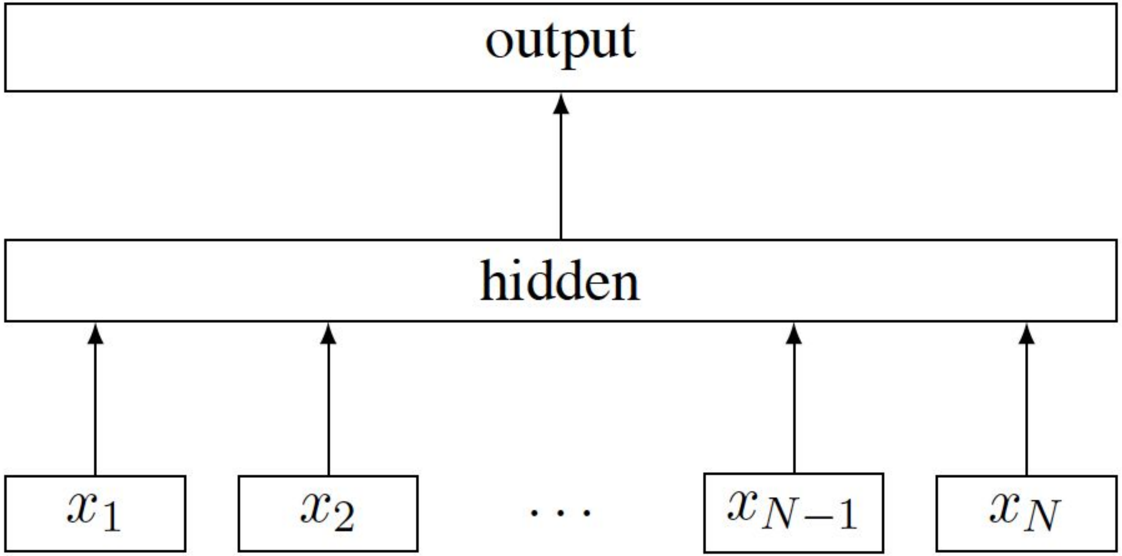
- Web search intent can be classified into 3 class: [Broder 2002]
  - **Navigational**. The immediate intent is to reach a particular site.
    - Greyhound Bus. Probable target http://www.greyhound.com
  - **Informational**. The intent is to acquire some information assumed to be present on one or more web pages.
    - San Francisco
  - **Transactional**. The intent is to perform some web-mediated activity.
    - Shopping activities
- In conversational AI, intent is usually task-specific, e.g.
  - I would like to book a flight from SFO to ANC: **FlightBooking**
  - What software can I use to view epub documents: **SoftwareRecommendation**.
- We focus on the latter in this tutorial.

Intent Classification - Methods

- Traditional methods
  - Features: bag of words, n-gram, TF-IDF.
  - Models: logistic regression, naive Bayes, SVM, random forest.
- Deep learning methods
  - Word embedding + linear classifier (fastText) [Joulin et. al. 2016].
  - Convolutional neural networks [Hashemi 2016] .
  - Bi-RNN + attention (joint slot filling) [Liu et al.2016] .

Intent Classification - fastText

[Joulin et. al. 2016]



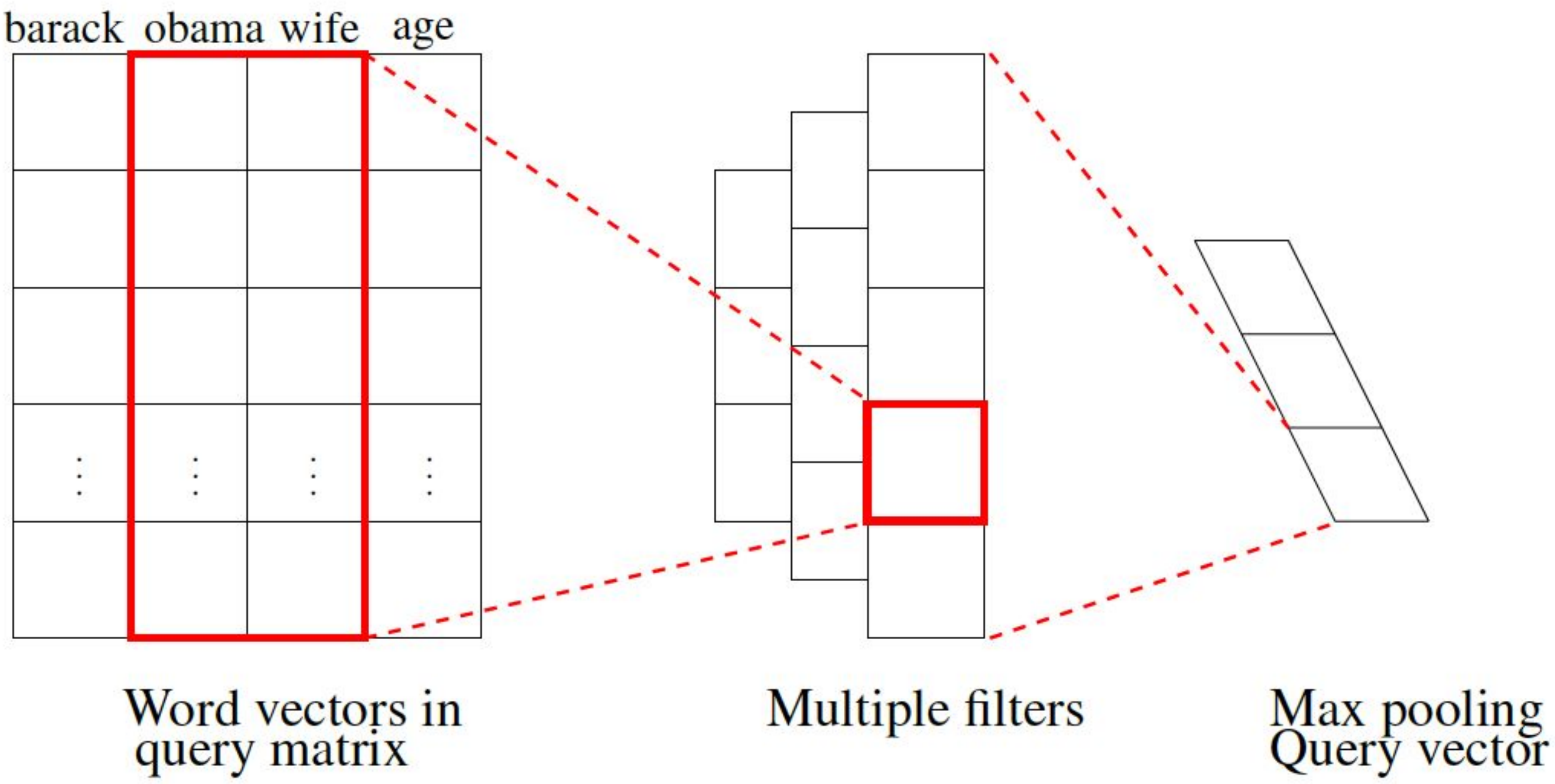
softmax

Average

word and N gram embedding features

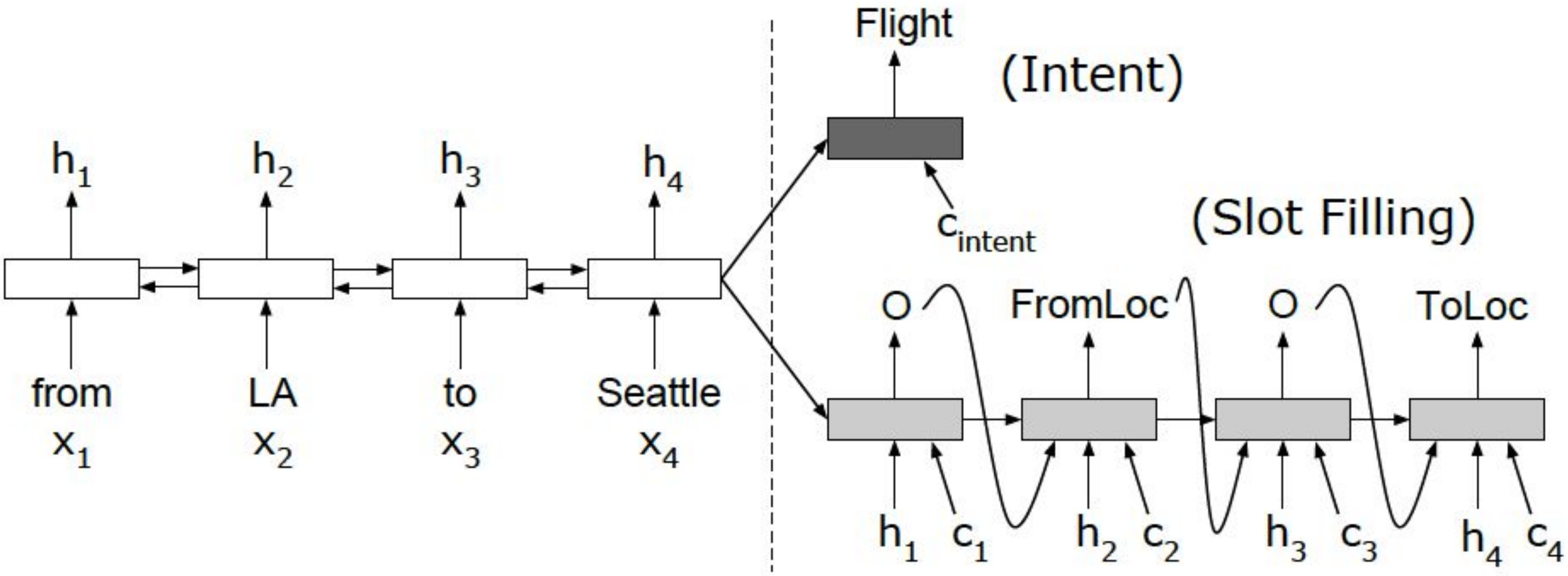
Intent Classification - CNN

[Hashemi 2016]



Intent Classification - Bi-RNN+Attention

[Liu et. al. 2016]



References - Intention Classification

- [Broder 2002] A. Broder. A taxonomy of web search. SIGIR Forum, 36(2):3–10, 2002.
- [Kim 2014] Y. Kim. Convolutional neural networks for sentence classification. In Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP), pages 1746–1751, Doha, Qatar, October 2014. Association for Computational Linguistics.
- [Joulin et. al. 2016] Joulin, A., Grave, E., Bojanowski, P., and Mikolov, T. (2017). Bag of tricks for efficient text classification. In Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics (EACL).
- [Lai et al.2015] Siwei Lai, Liheng Xu, Kang Liu, and Jun Zhao. 2015. Recurrent convolutional neural networks for text classification. In AAAI, pages 2267–2273
- [Liu et al.2016] Pengfei Liu, Xipeng Qiu, and Xuanjing Huang. 2016. Recurrent neural network for text classification with multi-task learning. arXiv preprint arXiv:1605.05101
- [Zhou et al.2016] Peng Zhou, Wei Shi, Jun Tian, Zhenyu Qi, Bingchen Li, Hongwei Hao, and Bo Xu. 2016. Attention-based bidirectional long short-term memory networks for relation classification. In The 54th Annual Meeting of the Association for Computational Linguistics, page 207.
- [Kim et. al. 2016] Joo-Kyung Kim, Gokhan Tur, Asli Celikyilmaz, Bin Cao, and Ye-Yi Wang. 2016. Intent detection using semantically enriched word embeddings. In Proceedings of SLT.
- [Hashemi 2016] Homa B Hashemi, Amir Asiaee, and Reiner Kraft. 2016. Query intent detection using convolutional neural networks. In International Conference on Web Search and Data Mining, Workshop on Query Understanding.
- [Liu et. al. 2016]. Liu and I. Lane, “Attention-based Recurrent Neural Network Models for Joint Intent Detection and Slot Filling,” in Interspeech, 2016.
- [Shi et. al. 2016] Y. Shi, K. Yao, L. Tian, and D. Jiang, “Deep lstm based feature mapping for query classification,” in Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, 2016, pp. 1501–1511.

Deep NLP in Search and Recommender Systems

- Language Understanding
  - Entity Tagging: word level prediction
  - Entity Disambiguation: knowledge base entity prediction
  - Intent Classification: sentence level prediction
  - **Sentiment Analysis and Opinion Mining**
- Document Retrieval and Ranking
  - Efficient Candidate Retrieval
  - Deep Ranking Models
- Language Generation for Assistance
  - Auto Completion
  - Query Suggestion
  - Spell Correction
  - Conversational Recommendation

Sentiment Analysis and Opinion Mining

- Problem Statement
- Challenges
- Traditional methods
- Deep learning methods

Sentiment Analysis - Problem Statement

- Interchangable with Opinion Mining in this tutorial.
- sentiment analysis mainly studies opinions which express or imply positive or negative sentiments [Liu 2012]
- Granularity:
  - Document level
  - Sentence level
  - Aspect level

Posted by: John Smith Date: September 10, 2011

“(1) I bought a Canon G12 camera six months ago. (2) I simply love it. (3) The **picture quality** is amazing. (4) The **battery life** is also long. (5) However, my wife thinks it is too heavy for her.”

Sentiment Analysis - Challenges

- Word ambiguity
  - This camera *sucks*.
  - This vacuum cleaner really *sucks*.
- Sentiment-less sentiment words
  - Can you tell me which camera is good?
- Sarcasm
  - What a great car! It stopped working in two days.
- Implicit sentiment
  - This washer uses a lot of water.

Traditional Methods

- Document/Sentence Sentiment Analysis
  - unsupervised: rule based, sentiment orientation
  - supervised: naive Bayes, logistic regression, SVM
- Aspect-based Sentiment Analysis
  - Aspect extraction
    - Frequent nouns and noun phrases
    - Opinion and target relations
    - Sequential tagging: HMM, CRF
    - Topic modeling
  - Aspect sentiment classification
    - Similar methods in document/sentence analysis.
    - limited to aspect dependent features

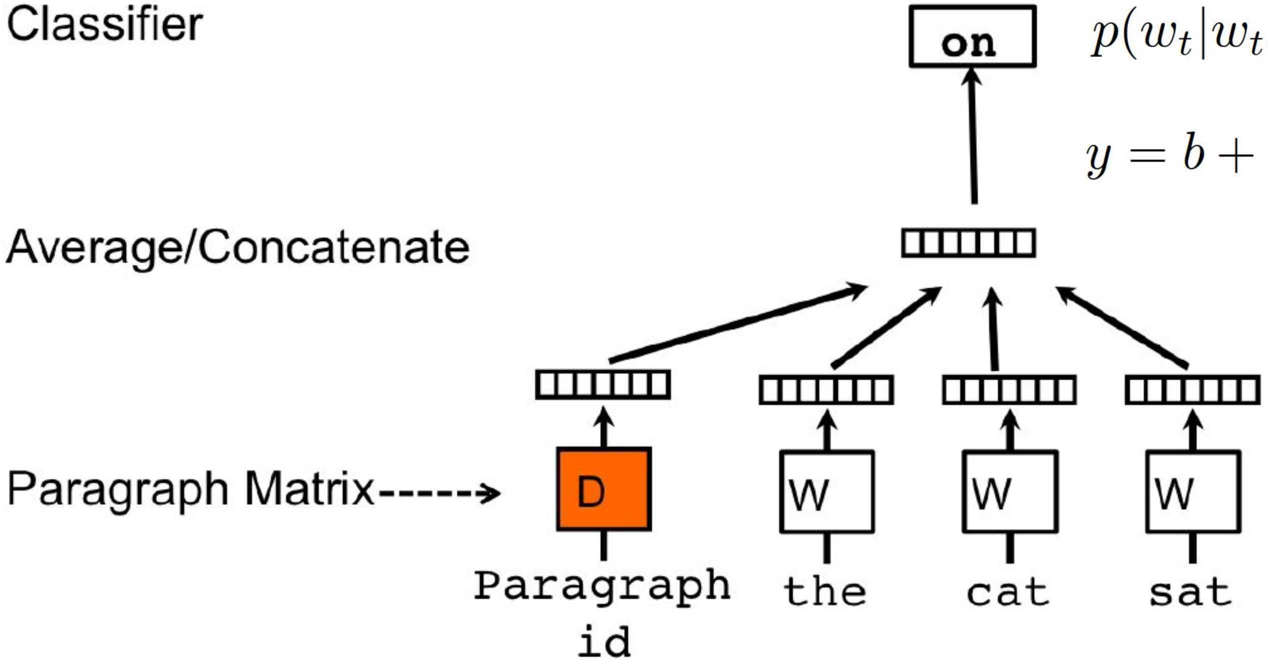
The **screen** is very **clear** but the **battery life** is too **short**.

Deep Learning Approaches

- Document/Sentence Sentiment Analysis
  - Use embedding alone
  - Use CNN/RNN
- Aspect-based Sentiment Analysis
  - Aspect extraction
    - CNN or RNN
  - Aspect sentiment analysis
    - attention-based RNN, memory networks

The **screen** is very **clear** but the **battery life** is too **short**.

Document Sentiment Analysis - Paragraph Vector

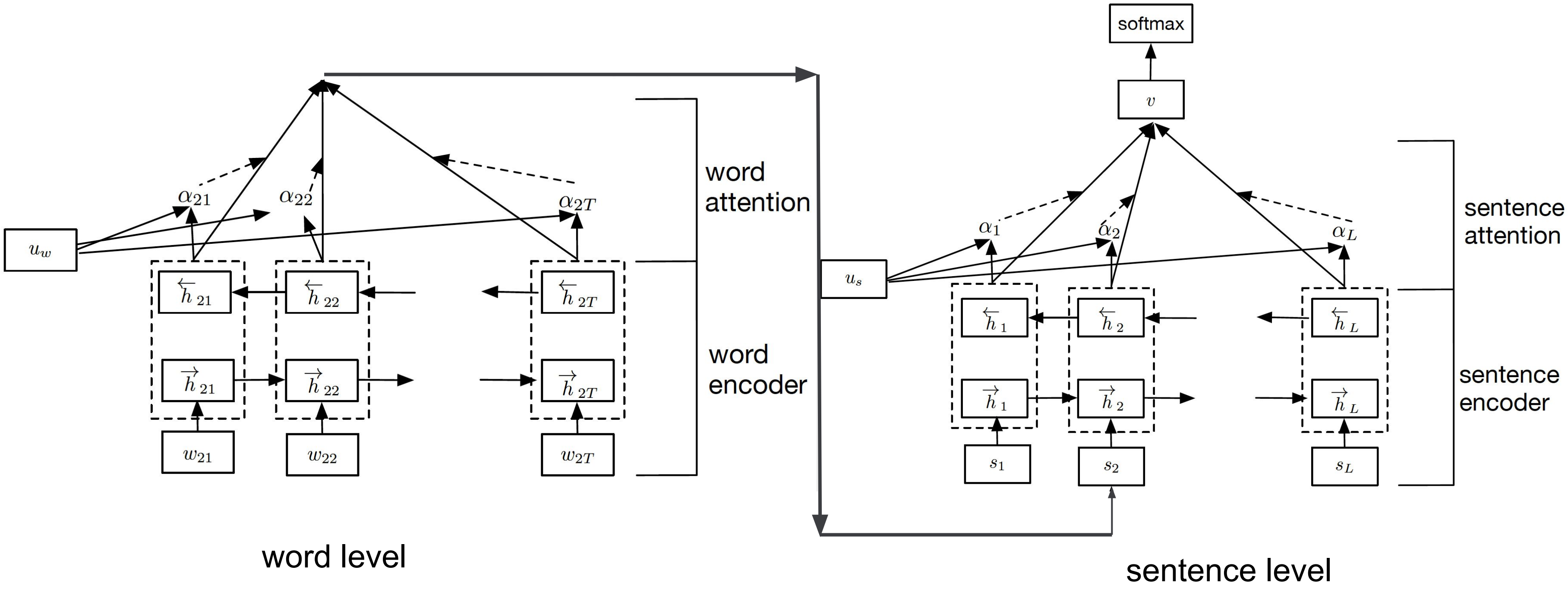


$$p(w_t|w_{t-k},\dots,w_{t+k}) = \frac{e^{y_{w_t}}}{\sum_i e^{y_i}}$$

$$y = b + Uh(w_{t-k},\dots,w_{t+k};W)$$

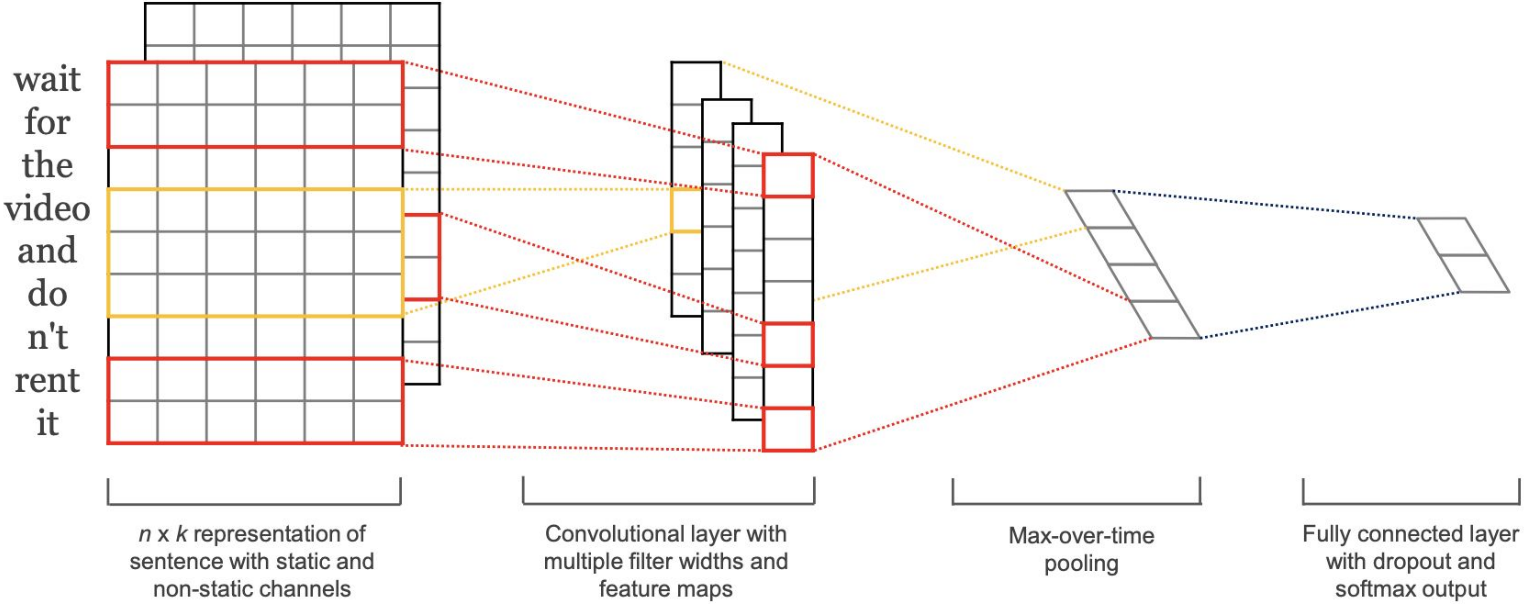
[Le et. al. 2014]

Document Sentiment Analysis - RNN



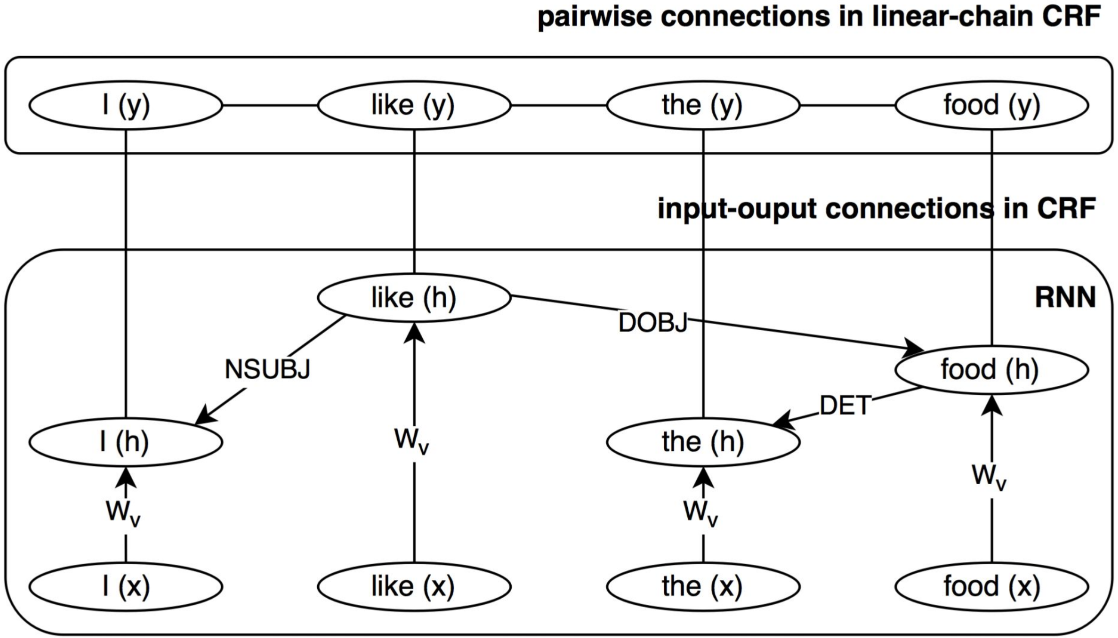
[Yang et. al. 2014]

Sentence Sentiment Analysis - CNN



[Kim 2014]

Aspect and Opinion Extraction - RNN



$$
\begin{aligned}
h_{\mathrm{I}} &= f(W_v \cdot x_{\mathrm{I}} + b), \\
h_{\mathrm{the}} &= f(W_v \cdot x_{\mathrm{the}} + b),
\end{aligned}
$$

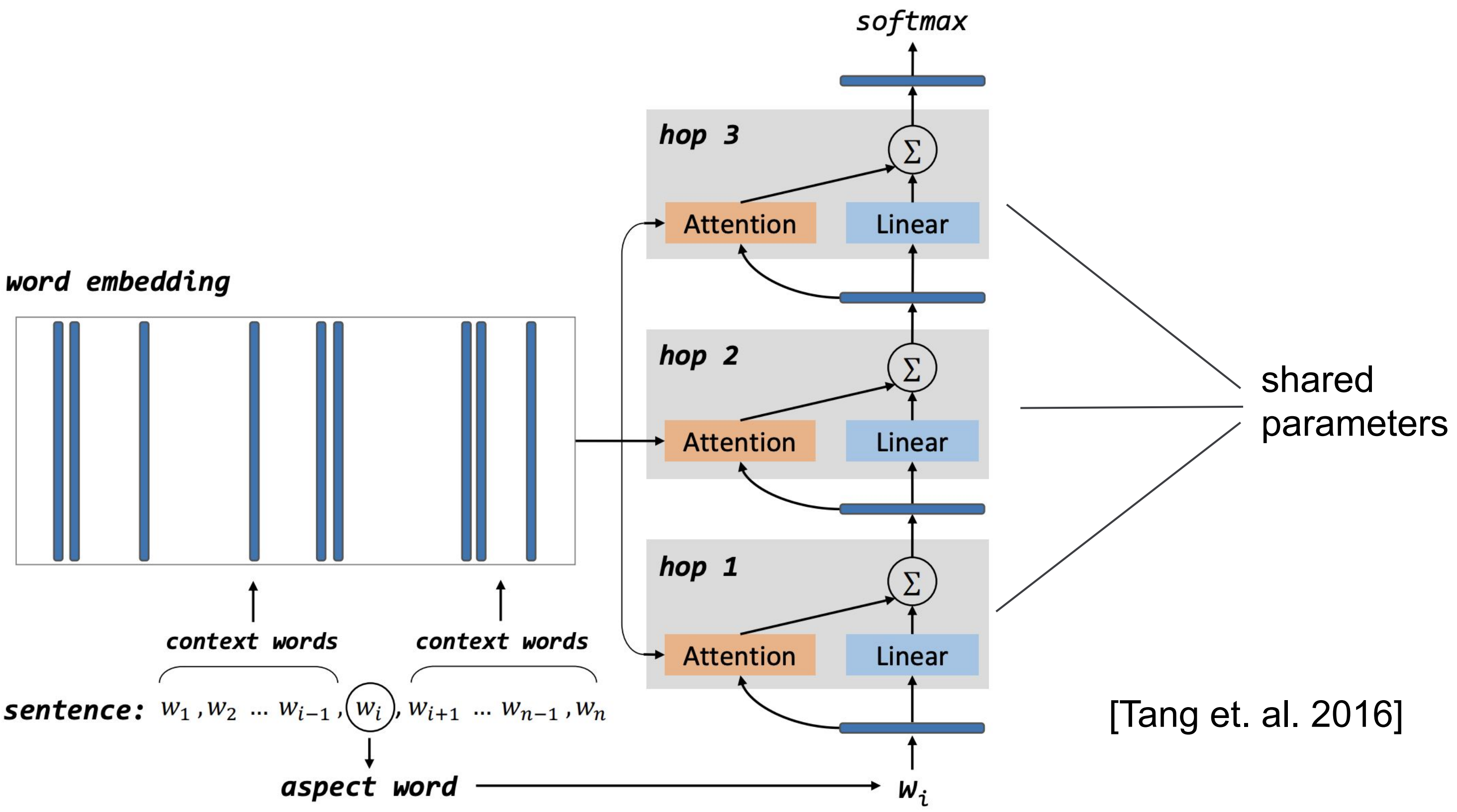
$$
\begin{aligned}
h_{\mathrm{food}} &= f(W_v \cdot x_{\mathrm{food}} + W_{\mathrm{DET}} \cdot h_{\mathrm{the}} + b), \\
h_{\mathrm{like}} &= f(W_v \cdot x_{\mathrm{like}} + W_{\mathrm{DOBJ}} \cdot h_{\mathrm{food}} \\
&\quad + W_{\mathrm{NSUBJ}} \cdot h_{\mathrm{I}} + b).
\end{aligned}
$$

Labels:

- BA (beginning of aspect)
- IA (inside of aspect),
- BO (beginning of opinion)
- IO (inside of opinion)
- O (others)

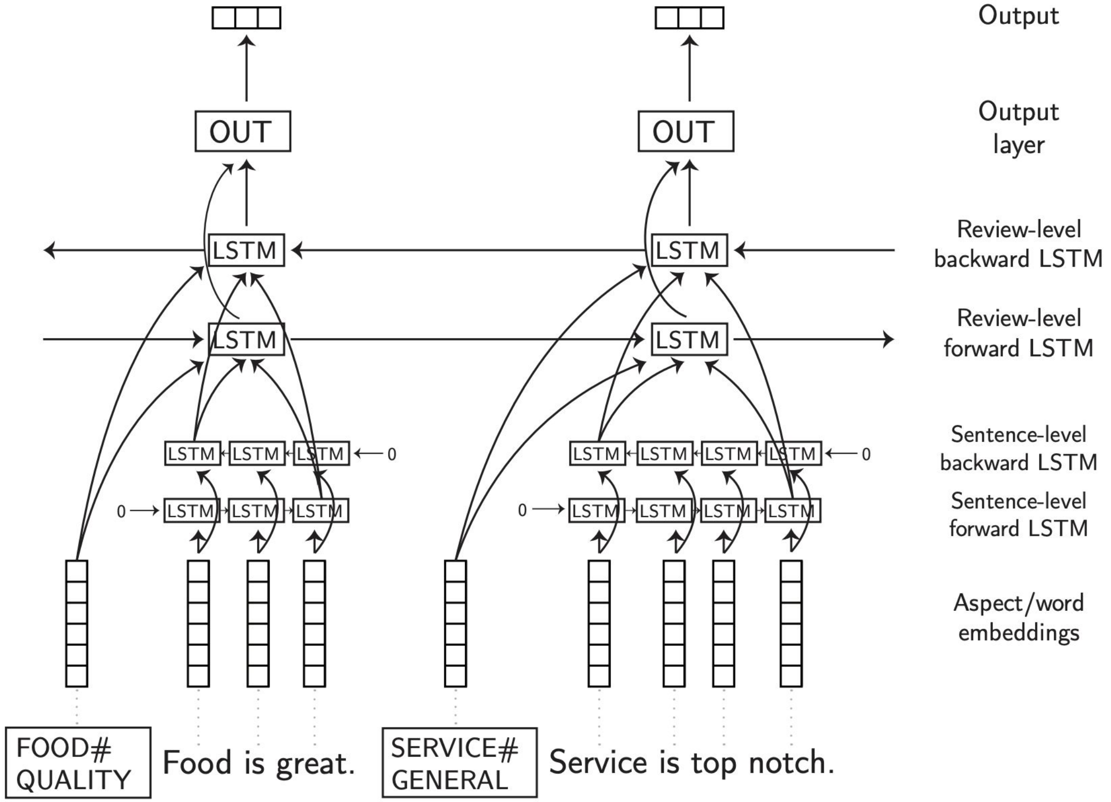
[Wang et. al. 2016]

Aspect Level Sentiment Analysis - Memory Networks



[Tang et. al. 2016]

Aspect Level Sentiment Analysis - Hierarchical RNN



- one aspect per sentence
- aspects are given
- Utilize the relationship between aspects

[Ruder et. al. 2016]

References - Sentiment Analysis and Opinion Mining

- [Kim 2014] Y. Kim. Convolutional neural networks for sentence classification. In Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP), pages 1746–1751, Doha, Qatar, October 2014. Association for Computational Linguistics.
- [Le et. al. 2014] Le Q, Mikolov T. Distributed representations of sentences and documents. In Proceedings of the International Conference on Machine Learning (ICML 2014), 2014.
- [Liu 2015] Liu B. Sentiment analysis: mining opinions, sentiments, and emotions. The Cambridge University Press, 2015.
- [Liu 2012] Liu B. Sentiment analysis and opinion mining (introduction and survey), Morgan & Claypool, May 2012.
- [Ruder et al. 2016] Sebastian Ruder, Parsa Ghaffari, and John G Breslin. A hierarchical model of reviews for aspect-based sentiment analysis. Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing, November, 2016.
- [Schouten et. al. 2016] Kim Schouten and Flavius Frasincar. 2016. Survey on Aspect-Level Sentiment Analysis. IEEE Trans. Knowledge and Data Engineering, 28(3):813-830
- [Tang et. al. 2015] Duyu Tang, Bing Qin, and Ting Liu. 2015. Document modeling with gated recurrent neural network for sentiment classification. In Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing, pages 1422–1432
- [Tang et. al. 2016] Tang D, Qin B, and Liu T. Aspect-level sentiment classification with deep memory network. arXiv preprint
- arXiv:1605.08900, 2016.
- [Xu et. al. 2016] Xu J, Chen D, Qiu X, and Huang X. Cached long short-term memory neural networks for document-level sentiment classification. In Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP 2016), 2016.
- [Yang et. al. 2016] Zichao Yang, Diyi Yang, Chris Dyer, Xiaodong He, Alex Smola, and Eduard Hovy. 2016. Hierarchical attention networks for document classification. In Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics
- [Zhang et. al. 2018] Lei Zhang, Shuai Wang, and Bing Liu. 2018. Deep learning for sentiment analysis: A survey. In Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery, page e1253. Wiley Online Library

References (Continued)

- [Poria et al.2016] Soujanya Poria, Erik Cambria, and Alexander Gelbukh. 2016a. Aspect extraction for opinion mining with a deep convolutional neural network. Knowledge-Based Systems, 108:42–49.
- [Wang et. al. 2016] Wang W, Pan SJ, Dahlmeier D, and Xiao X. Recursive neural conditional random fields for aspect-based sentiment analysis. In Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP 2016), 2016.

Deep NLP in Search and Recommender Systems - Document Retrieval & Ranking

Jun Shi, Weiwei Guo

Deep NLP in Search and Recommender Systems

- Language Understanding
  - Entity Tagging: word level prediction
  - Entity Disambiguation: knowledge base entity prediction
  - Intent Classification: sentence level prediction
  - Sentiment Analysis and Opinion Mining
- **Document Retrieval and Ranking**
  - **Efficient Candidate Retrieval**
  - **Deep Ranking Models**
- Language Generation for Assistance
  - Auto Completion
  - Query Suggestion
  - Spell Correction
  - Conversational Recommendation

Document Retrieval and Ranking

- Efficient Candidate Retrieval
- Deep Neural Ranking

Efficient Candidate Retrieval

- Syntactic retrieval
  - based on string matching
  - use inverted index
  - can include different fields (name, title, etc).
- Semantic retrieval
  - based on vector space representation.
  - approximate nearest neighbor search

Syntactic Retrieval

Query: mike software engineer

Inverted index

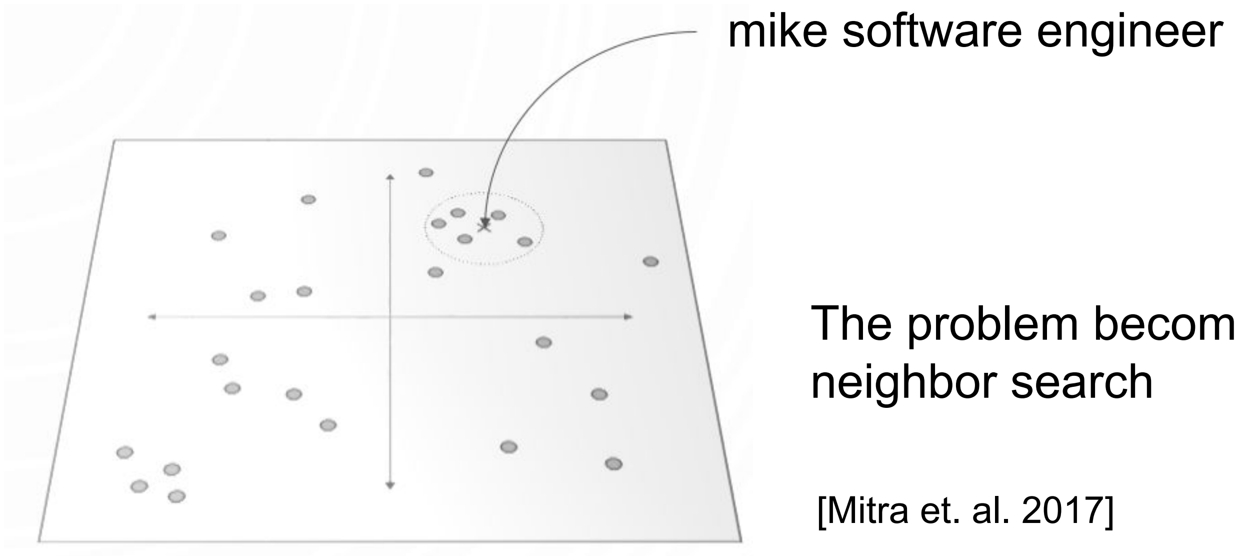
mike	Doc1, **Doc2**, **Doc4**, ...
software	**Doc2**, Doc3, **Doc4**, Doc7, ...
engineer	Doc1, **Doc2**, **Doc4**, Doc9, ...

Results: **Doc2, Doc4**

It won't retrieve a document contains "mike is a software developer,..."

Semantic Retrieval - Concept

mike software engineer $\xrightarrow{\text{embedding}}$ [0.3, 1.27, -3.4, ...]



The problem becomes nearest neighbor search

[Mitra et. al. 2017]

Semantic Retrieval - Vector Generation

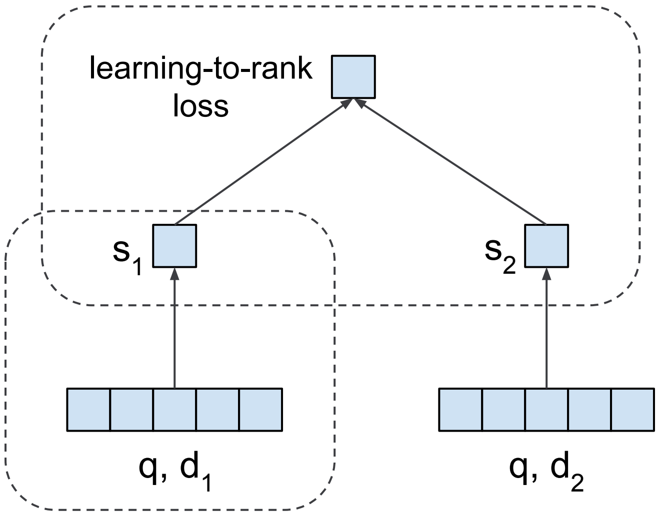
- Bag of words
  - count based, TF-IDF vectors.
- Embedding vectors
  - word embedding: word2vec, Glove, BERT, etc.
  - sentence/document embedding
    - universal sentence encoder [Cer et. al. 2018]
    - Gaussian document representation [Giannis et. al. 2017]
    - power mean concatenation [Rücklé et. al. 2018]

Deep NLP in Search and Recommender Systems

- Language Understanding
  - Entity Tagging: word level prediction
  - Entity Disambiguation: knowledge base entity prediction
  - Intent Classification: sentence level prediction
  - Sentiment Analysis and Opinion Mining
- Document Retrieval and Ranking
  - Efficient Candidate Retrieval
  - **Deep Ranking Models**
- Language Generation for Assistance
  - Auto Completion
  - Query Suggestion
  - Spell Correction
  - Conversational Recommendation

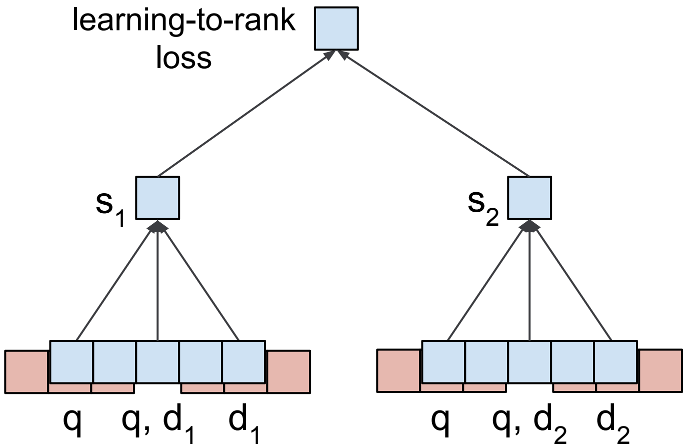
Deep Neural Ranking - Agenda

- Traditional methods
  - Ranking features
  - Learning to rank



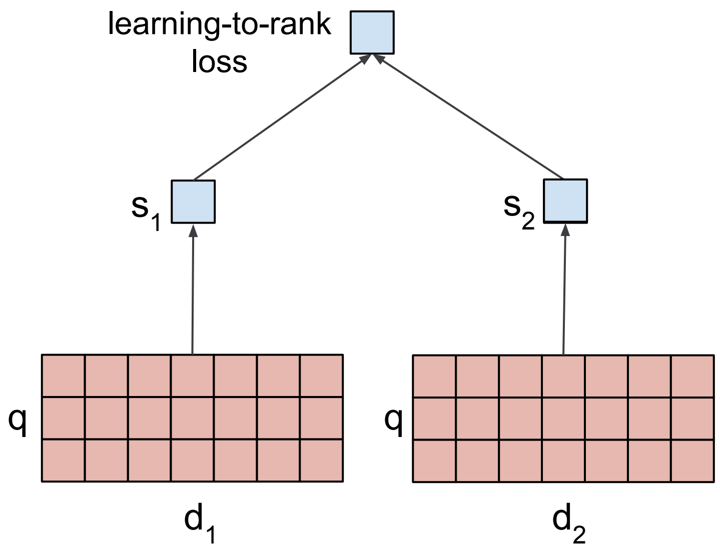
Deep Neural Ranking - Agenda

- Traditional methods
  - Ranking features
  - Learning to rank
- Deep neural ranking
  - Siamese Networks



Deep Neural Ranking - Agenda

- Traditional methods
  - Ranking features
  - Learning to rank
- Deep neural ranking
  - Siamese Networks
  - Interaction-based Networks



Deep Neural Ranking - Agenda

- Traditional methods
  - Ranking features
  - Learning to rank
- Deep neural ranking
  - Siamese Networks
  - Interaction-based Networks
  - Wide & Deep Model

Traditional Ranking Features

- Hand-crafted features
  - Query/document matching features
    - Cosine similarity between query and doc title
    - Entity matching
    - Clickthrough rate from query to this doc based on search log
    - …….
  - Document alone
    - popularity
    - number of incoming links
    - ……

Learning to Rank

(Burges, 2010)

- Pointwise ranking
  - Logistic regression

$$\frac{1}{1+e^{-s}} \quad \text{for } y = 1$$

- Pairwise ranking

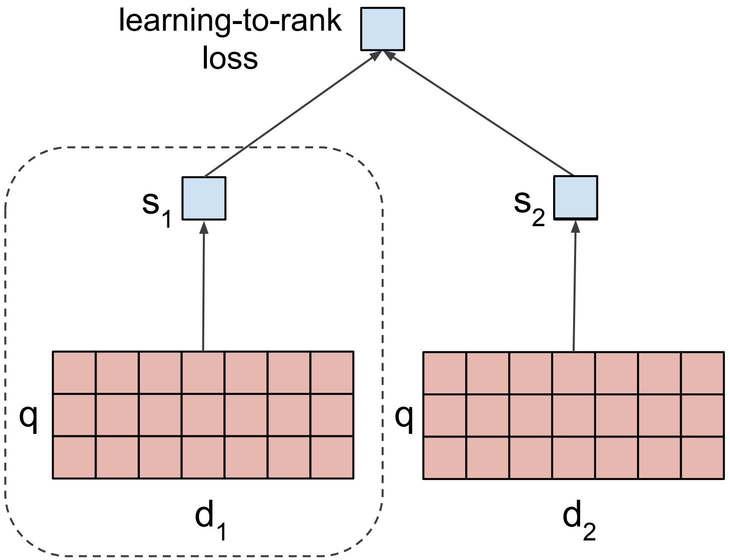
$$\frac{1}{1+e^{-(s_1-s_2)}} = \frac{e^{s_1}}{e^{s_1}+e^{s_2}}$$

- Listwise ranking
  - Cross entropy

$$\sum_i y_i \cdot \frac{e^{s_i}}{e^{s_1}+e^{s_2}+\ldots+e^{s_n}}$$

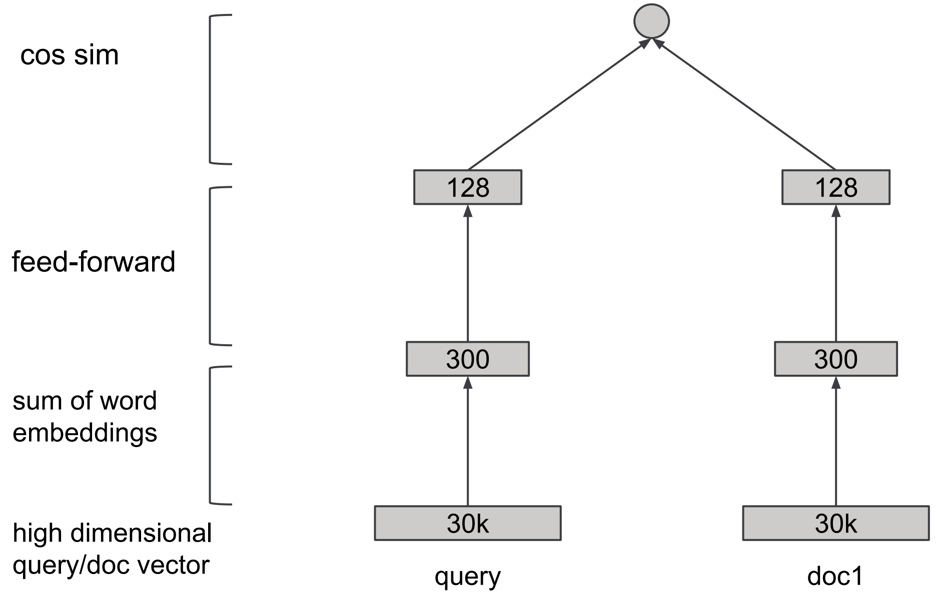
Deep Neural Ranking

- Focus on computing query/document score
- Two categories:
  - Siamese Networks
  - Interaction based Networks



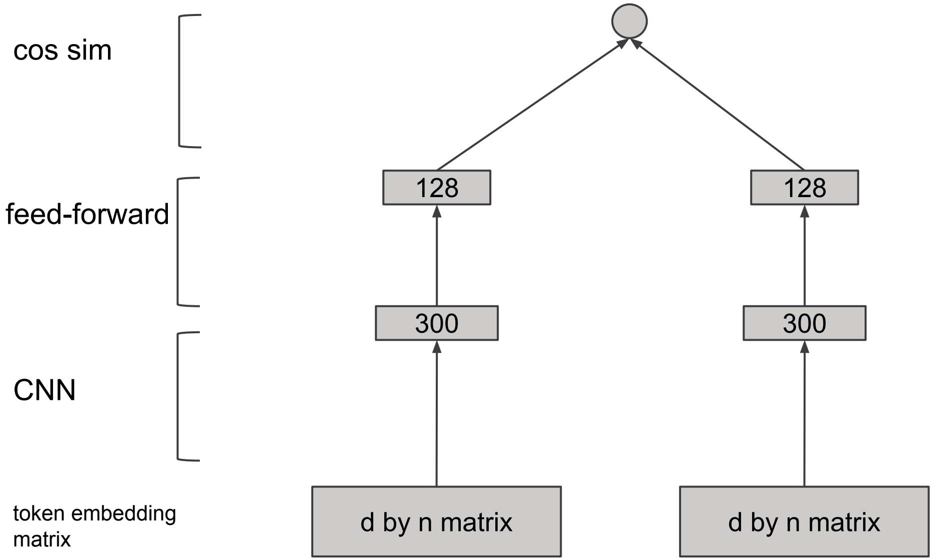
Deep Structured Semantic Model

(Huang et al., 2013)



Modeling Word Sequence by CNN

(Shen et al., 2014)

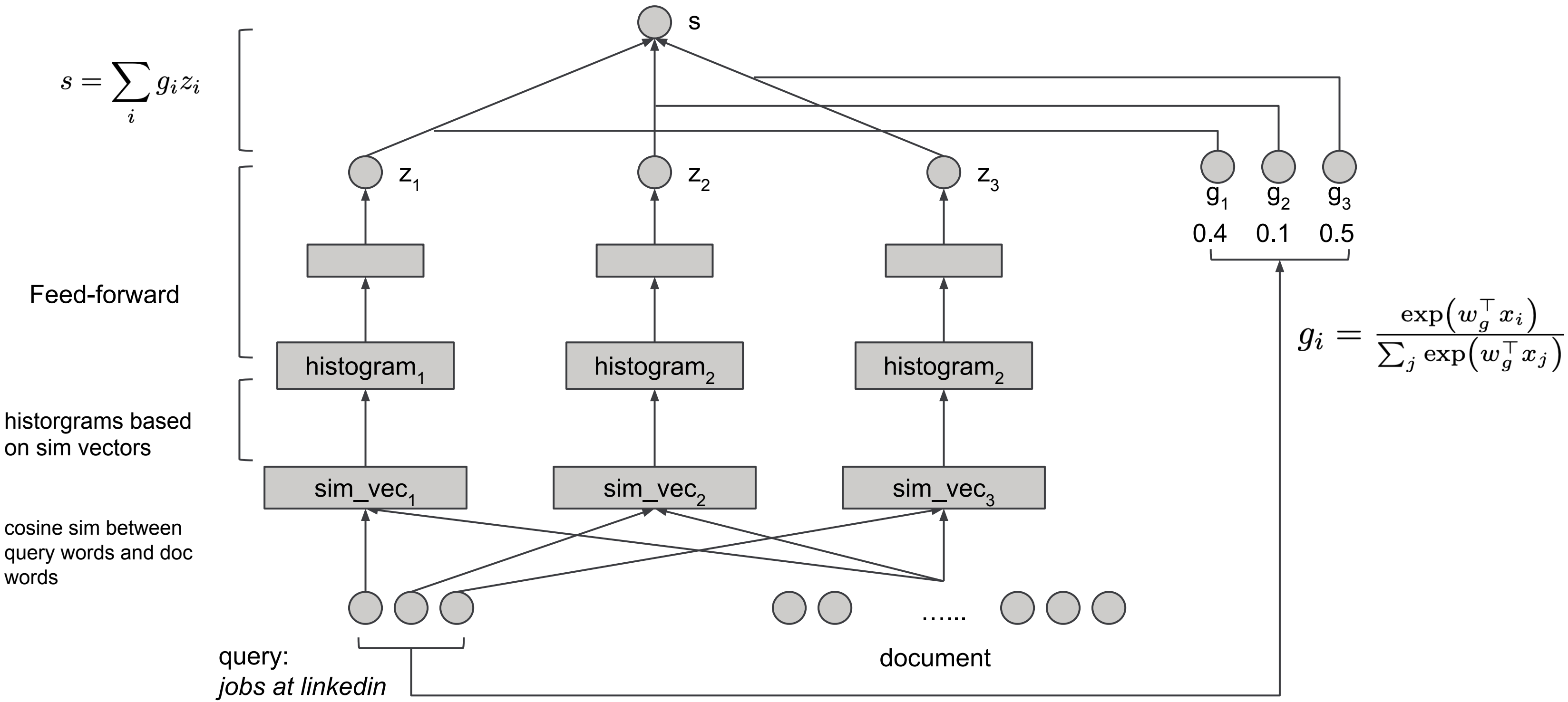


Siamese Networks

- Pros:
  - Generalization; semantic matching
  - Efficient; doc embs can be precomputed
- Cons:
  - Lexical features lost: people/company names, rare words

Interaction-based Networks

(Guo et al., 2016)



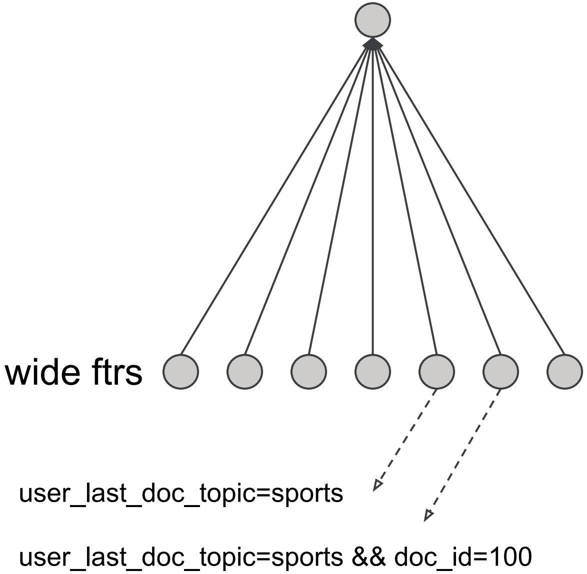
Search Ranking Neural Networks Summary

- Only focus on query/doc scoring
- End-to-end models
- Two popular architectures

	Siamese Network	**Interaction Network**
Match	topical matches	lexical matches
Latency	small	large

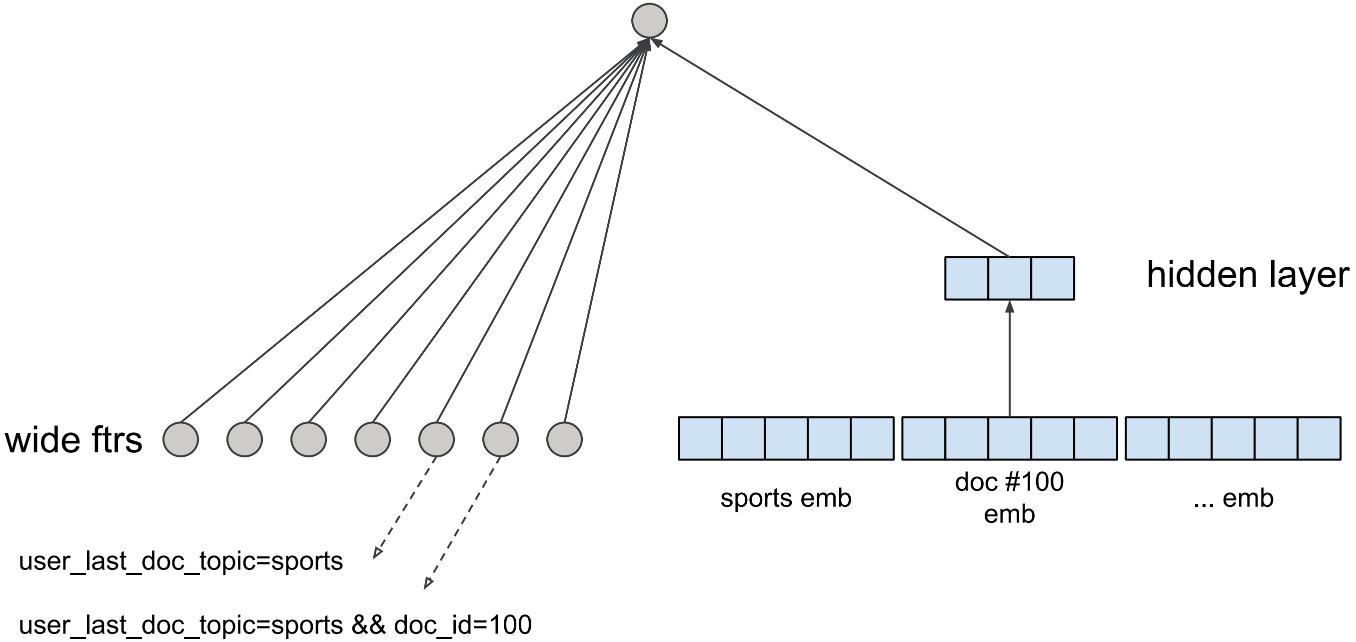
Recommender System: Wide & Deep Model

(Cheng et al., 2016)



Recommender System: Wide & Deep Model

(Cheng et al., 2016)



References - Deep Neural Ranking

- Huang, Po-Sen, Xiaodong He, Jianfeng Gao, Li Deng, Alex Acero, and Larry Heck. "Learning deep structured semantic models for web search using clickthrough data." In Proceedings of the 22nd ACM international conference on Information & Knowledge Management, pp. 2333-2338. ACM, 2013.
- Shen, Yelong, Xiaodong He, Jianfeng Gao, Li Deng, and Grégoire Mesnil. "A latent semantic model with convolutional-pooling structure for information retrieval." In Proceedings of the 23rd ACM international conference on conference on information and knowledge management, pp. 101-110. ACM, 2014.
- Rodrigo Nogueira, Kyunghyun Cho, Passage re-ranking with BERT, 2019
- Burges. "From ranknet to lambdarank to lambdamart: An overview." *Learning.* 2010.
- Guo, Jiafeng, Yixing Fan, Qingyao Ai, and W. Bruce Croft. "A deep relevance matching model for ad-hoc retrieval." In *Proceedings of the 25th ACM International on Conference on Information and Knowledge Management*, pp. 55-64. ACM, 2016.
- Mitra, Bhaskar, Fernando Diaz, and Nick Craswell. "Learning to match using local and distributed representations of text for web search." In Proceedings of the 26th International Conference on World Wide Web, pp. 1291-1299. International World Wide Web Conferences Steering Committee, 2017.
- Cheng, Heng-Tze, Levent Koc, Jeremiah Harmsen, Tal Shaked, Tushar Chandra, Hrishi Aradhye, Glen Anderson et al. "Wide & deep learning for recommender systems." In Proceedings of the 1st workshop on deep learning for recommender systems, pp. 7-10. ACM, 2016.
- Xiong, Chenyan, Zhuyun Dai, Jamie Callan, Zhiyuan Liu, and Russell Power. "End-to-end neural ad-hoc ranking with kernel pooling." In Proceedings of the 40th International ACM SIGIR conference on research and development in information retrieval, pp. 55-64. ACM, 2017.
- Dai, Zhuyun, Chenyan Xiong, Jamie Callan, and Zhiyuan Liu. "Convolutional neural networks for soft-matching n-grams in ad-hoc search." In Proceedings of the eleventh ACM international conference on web search and data mining, pp. 126-134. ACM, 2018.

Deep NLP in Search and Recommender Systems - Language Generation for Assistance

Weiwei Guo

Deep NLP in Search and Recommender Systems

- Language Understanding
  - Entity Tagging: word level prediction
  - Entity Disambiguation: knowledge base entity prediction
  - Intent Classification: sentence level prediction
  - Sentiment Analysis and Opinion Mining
- Document Retrieval and Ranking
  - Efficient Candidate Retrieval
  - Deep Ranking Models
- **Language Generation for Assistance**
  - **Auto Completion**
  - **Query Reformulation**
  - **Spell Correction**
  - **Conversational Recommendation**

Language Generation for Assistance

- Auto Completion
- Query Reformulation
- Spell Correction
- Conversational Recommendation

Common

- Goal: improve user experience
- How: interact with users
- NLP: Language generation

Difference

- Character/word modeling
- Generation models:
  - language modeling, seq2seq

Deep NLP in Search and Recommender Systems

- Language Understanding
  - Entity Tagging: word level prediction
  - Entity Disambiguation: knowledge base entity prediction
  - Intent Classification: sentence level prediction
  - Sentiment Analysis and Opinion Mining
- Document Retrieval and Ranking
  - Efficient Candidate Retrieval
  - Deep Ranking Models
- Language Generation for Assistance
  - **Auto Completion**
  - Query Reformulation
  - Spell Correction
  - Conversational Recommendation

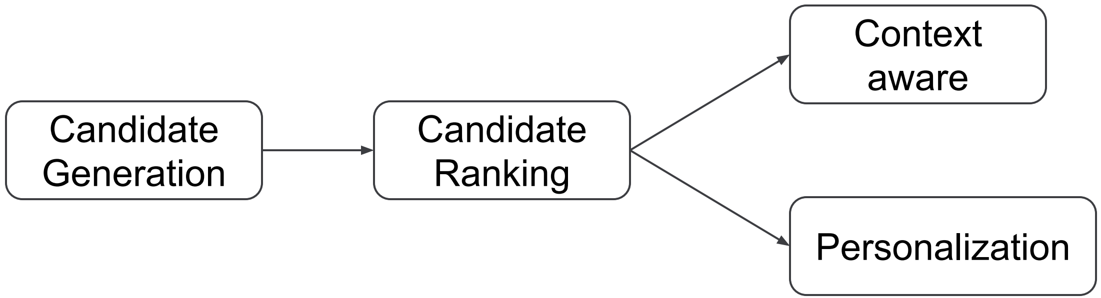
Query Auto-Completion

- Problem statement: save user keystrokes by predicting the entire query

Challenges

- Limited Context
  - Hard to extract features due to limited words in a query
- Latency
  - The search component with most strict requirement on latency

Agenda



Traditional methods

Candidate Generation

- Collect completed queries and associated frequency from search log
- Efficiently retrieve most frequent queries starting with the prefix
  - Using a trie data structure

soft $\xrightarrow{\text{candidate generation}}$

software engineer	100
softball	50
softmax	40
softbank	35
…...	

Candidate Generation for Rare Prefix

(Mitra & Craswell, 2015)

- No such prefix in search log
  - “*cheapest flights from seattle to*”
- Another trie on suffixes
  - “distance from seattle to dc” →
    “*from seattle to dc*”, “*seattle to dc”, “to dc”*

“*cheapest flights from seattle to*” → “*to*” →

to dc	100
to sfo	50
to airport	40
to seattle	35
…...	

Candidate Ranking

- Challenge: very few features can be extracted

soft	candidate generation →	*softbank*	candidate ranking →	*software engineer*	*100*
		softball		*soft skills*	*50*
		softmax		*softball*	*20*
		software engineer		*softbank*	*15*
		……		*……*	

frequency

Context-Aware

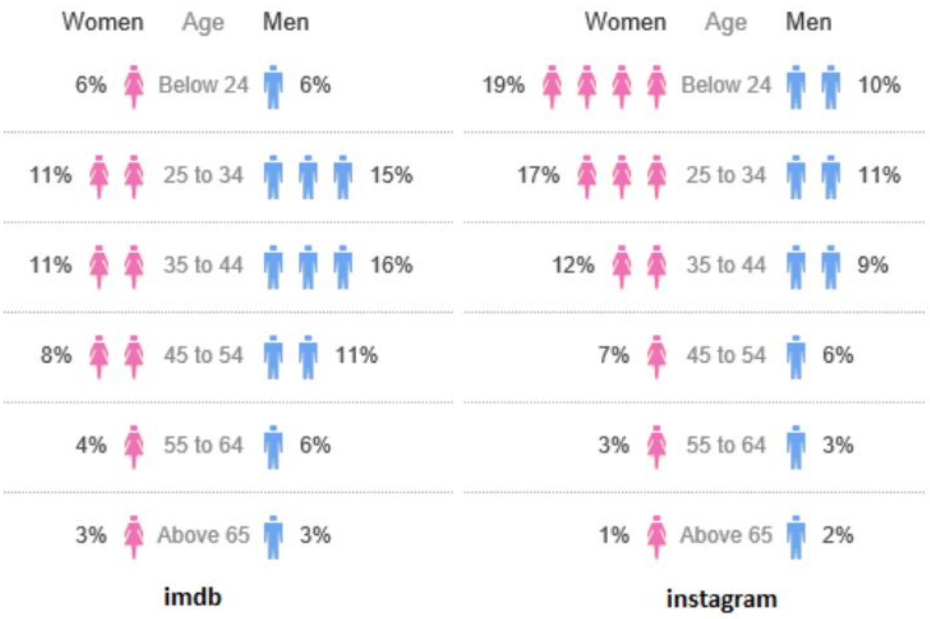
(Bar-Yossef & Kraus, 2011)

- Key idea: identify the candidates most similar to previous queries

	(candidates)			sim score
infant n →	*infant nike shoes*	→	*infant, nike, shoe, adidas, clothes...*	0.1
	infant nutrition		*infant, nutrition, baby, eat, food...*	0.8
	…...		…...	
	(previous queries)			
	baby eating disorder	→	*baby, eating, disorder, nutrition, food...*	
	…...		…...	

Personalization

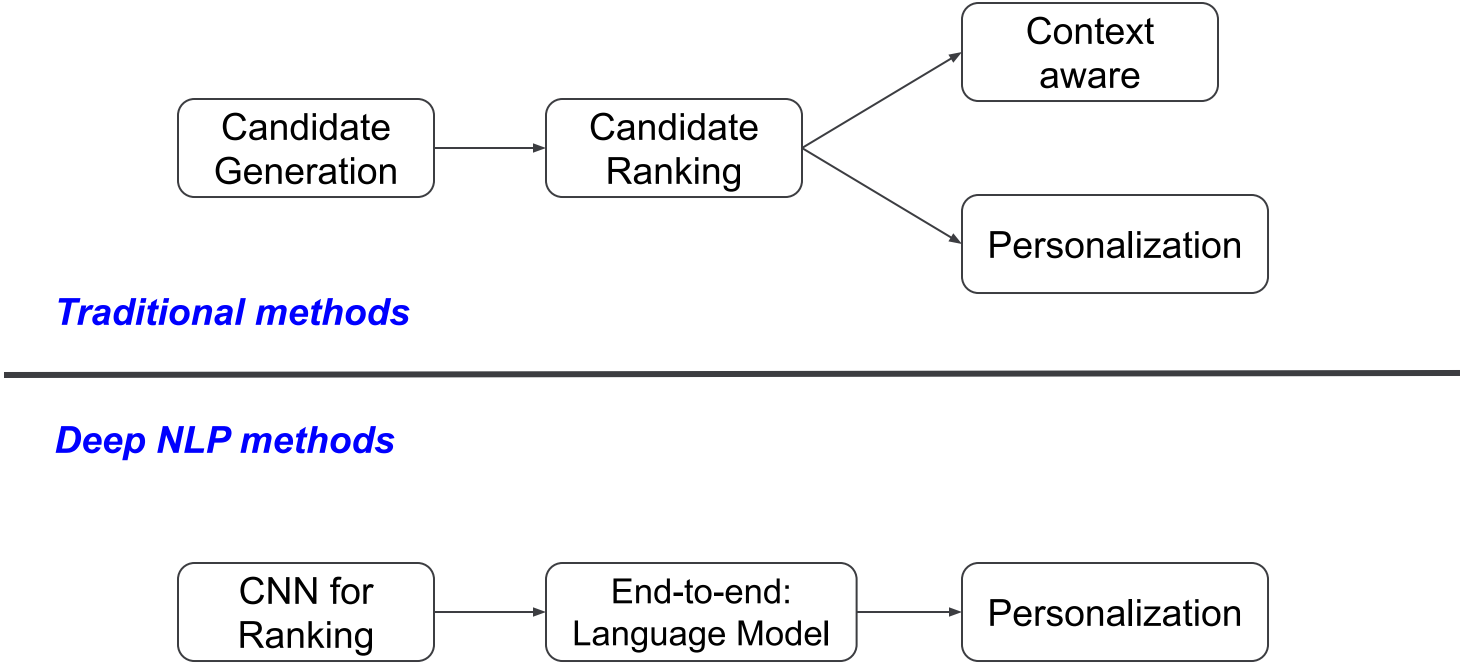
(Shokouhi, 2013)



query prefix is "i"

Feature list
Same age
Same gender
Same region
……

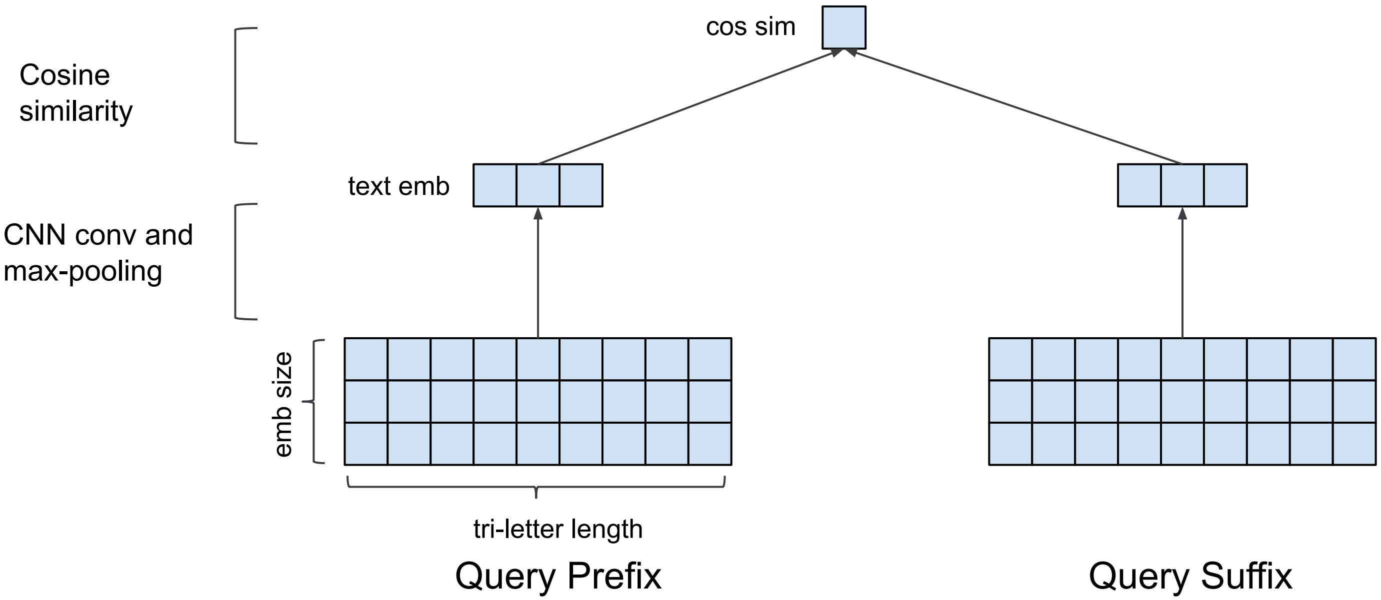
Agenda


Candidate Generation
Candidate Ranking
Context aware
Personalization
Traditional methods
Deep NLP methods
CNN for Ranking
End-to-end: Language Model
Personalization

Apply Deep Models in Ranking

(Mitra & Craswell, 2015)

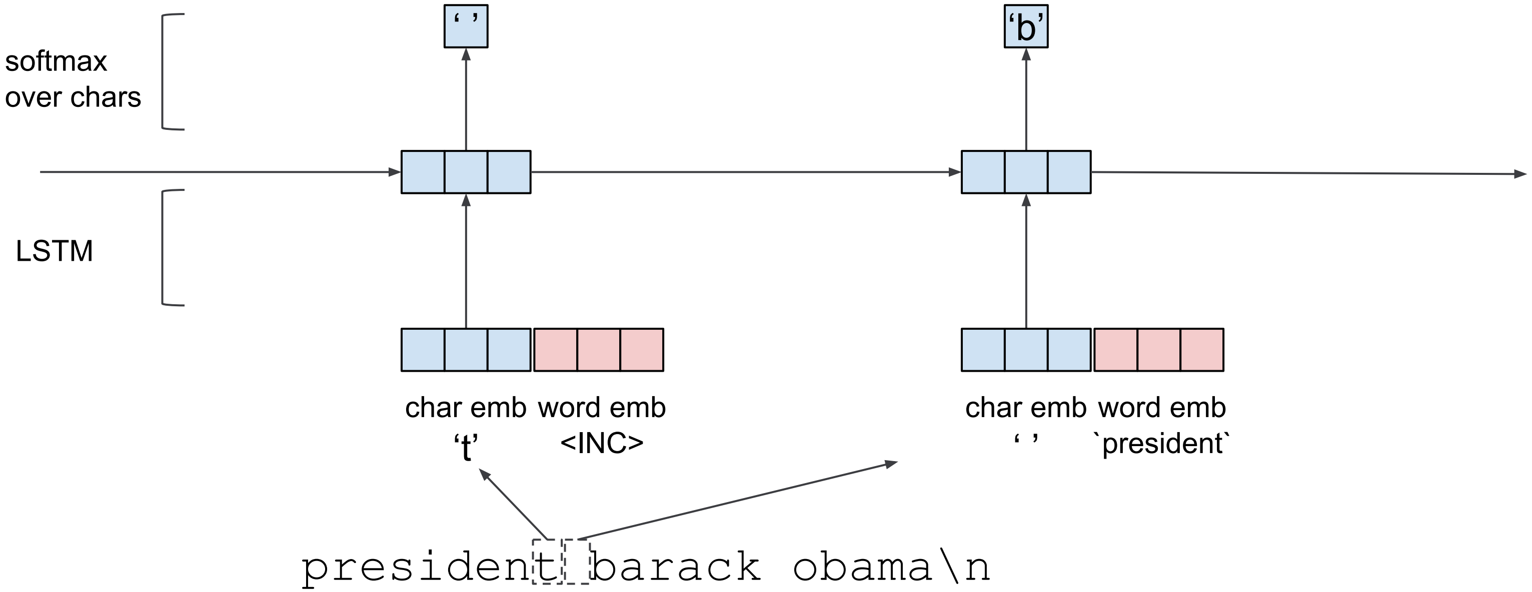
- Measuring the semantic coherence between prefix and suffix



Language Modeling for Auto-Completion

(Park & Chiba, 2017)

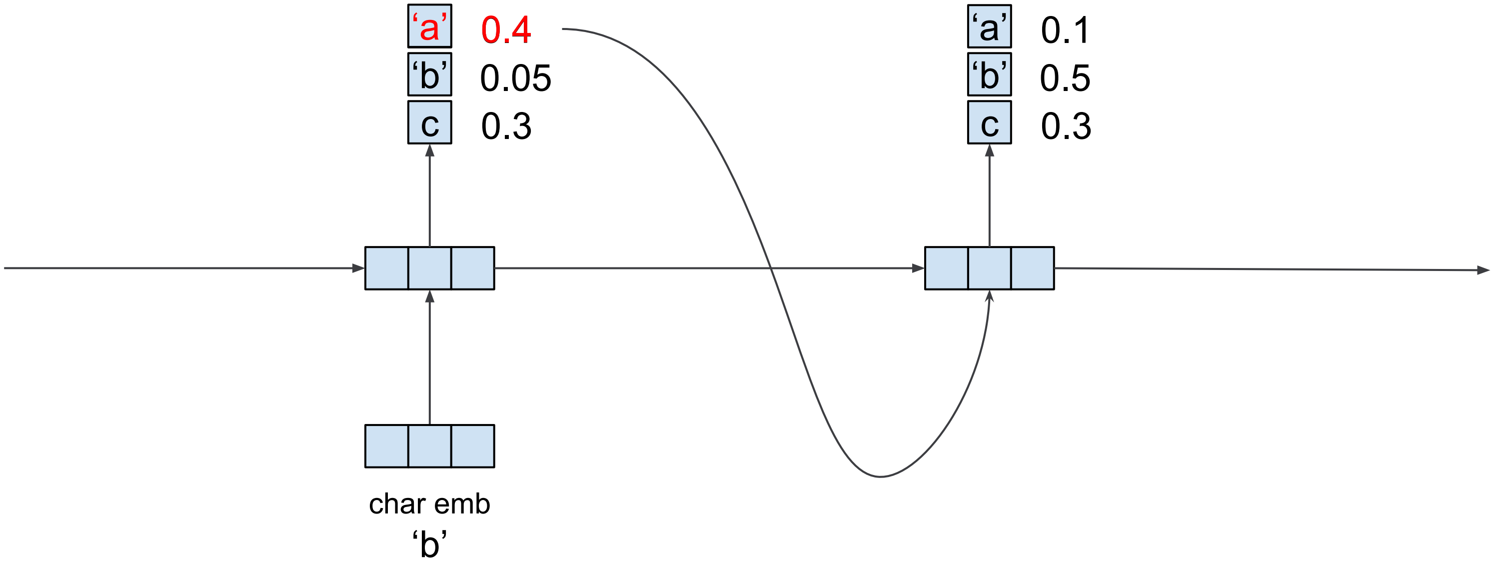
- Training: Character level language model + word embedding



Language Modeling for Auto-Completion

(Park & Chiba, 2017)

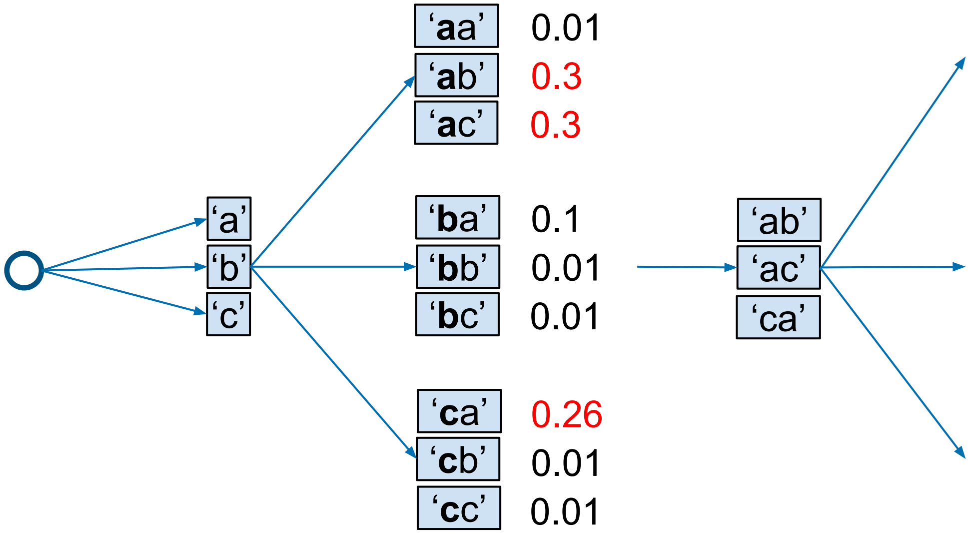
- Testing: Generating and ranking candidates at the same time



Language Modeling for Auto-Completion

(Park & Chiba, 2017)

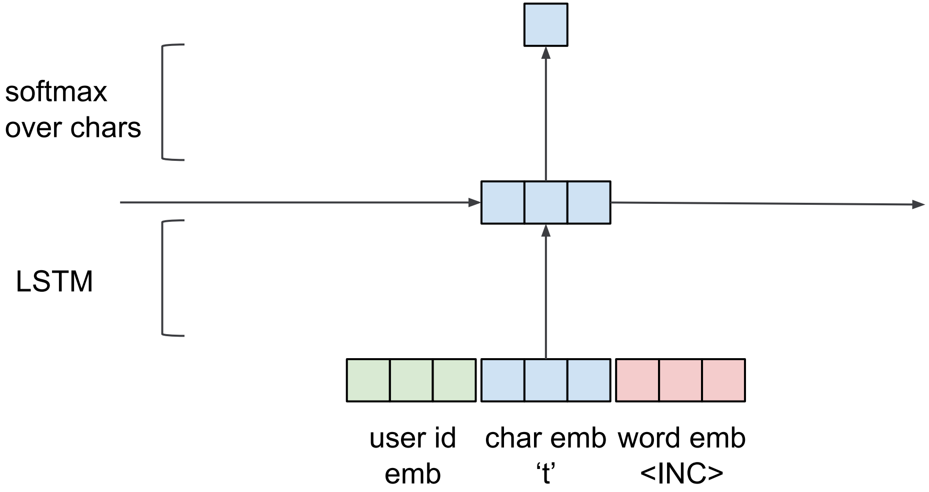
- Testing: Generating and ranking candidates at the same time
- ~~Greedy~~ vs beam search



Personalization

(Fiorin & Lu, 2018)

- Embeddings for User Ids



Query Auto-Completion: Summary

- Traditional methods: hard to extract features
- Deep language model framework:
  - Very flexible to incorporate personalized/contextualized information
  - An end-to-end solution
    - Train: all parameters are optimized together
    - Test: generation and ranking are performed at the same time
  - Cons
    - Time-consuming
    - May generate wrong words

Reference

- Mitra, Bhaskar, and Nick Craswell. "Query auto-completion for rare prefixes." In Proceedings of the 24th ACM international on conference on information and knowledge management, pp. 1755-1758. ACM, 2015.
- Park, Dae Hoon, and Rikio Chiba. "A neural language model for query auto-completion." In Proceedings of the 40th International ACM SIGIR Conference on Research and Development in Information Retrieval, pp. 1189-1192. ACM, 2017.
- Bar-Yossef, Ziv, and Naama Kraus. "Context-sensitive query auto-completion." In Proceedings of the 20th international conference on World wide web, pp. 107-116. ACM, 2011.
- Shokouhi, Milad. "Learning to personalize query auto-completion." In Proceedings of the 36th international ACM SIGIR conference on Research and development in information retrieval, pp. 103-112. ACM, 2013.
- Jaech, Aaron, and Mari Ostendorf. "Personalized language model for query auto-completion." arXiv preprint arXiv:1804.09661 (2018).
- Fiorini, Nicolas, and Zhiyong Lu. "Personalized neural language models for real-world query auto completion." arXiv preprint arXiv:1804.06439 (2018).

Deep NLP in Search and Recommender Systems

- Language Understanding
  - Entity Tagging: word level prediction
  - Entity Disambiguation: knowledge base entity prediction
  - Intent Classification: sentence level prediction
  - Sentiment Analysis and Opinion Mining
- Document Retrieval and Ranking
  - Efficient Candidate Retrieval
  - Deep Ranking Models
- Language Generation for Assistance
  - Auto Completion
  - **Query Reformulation**
  - Spell Correction
  - Conversational Recommendation

Query Reformulation

- Problem Statement: automatically reformulate the previous query

facebook developers | registration
https://go.fb.com/become-a-facebook-developer-lp05.html
Code to connect with 2B+ people with Facebook. Become a Facebook **Developer** to build and ship your application. Become a Facebook **Developer** ...

Developer Tools Archives - Facebook Code
https://code.fb.com/category/developer-tools/
Our mission is to increase **developer** efficiency so that we can continue to ship awesome products quickly. To accomplish this, we have focused on building a ...

Facebook Developer Conference. April 30 - May 1, 2019.San Jose, CA.
https://www.f8.com/
Facebook's annual **developer** conference spotlights our global community, the latest technology from our family of apps, and the future we are building together.

Best Facebook App Development Company, Best FB Developers
https://www.cygnismedia.com/social-media.../best-facebook-application.html
Effective branding is possible by attractive Facebook application development. Cygnis Media listed in Best App Development companies for Facebook. Hire best ...

Search results for "*fb developer*"

Searches related to fb developer

fb **developers support**	**facebook** developer **tutorial**
facebook developer **support**	**facebook** developer **alerts**
facebook for developers products	**facebook for** developer **tools**
facebook developer **ecosystem**	**facebook** developer **forum**

Reformulated queries for "*fb developer*"

Agenda

collaborative filtering

Traditional methods

Traditional Approach: Collaborative Filtering

(Rida et al., 2012)

- Collect query pairs issued by the same user from search log
- Treat each query as an ID, and build a query-to-query matrix
  - Value in the matrix is TF-IDF
- Issues:
  - No understanding of words in the query
  - Data sparsity
  - Cannot generate new queries

Agenda

collaborative filtering

Traditional methods

Deep NLP methods

seq2seq

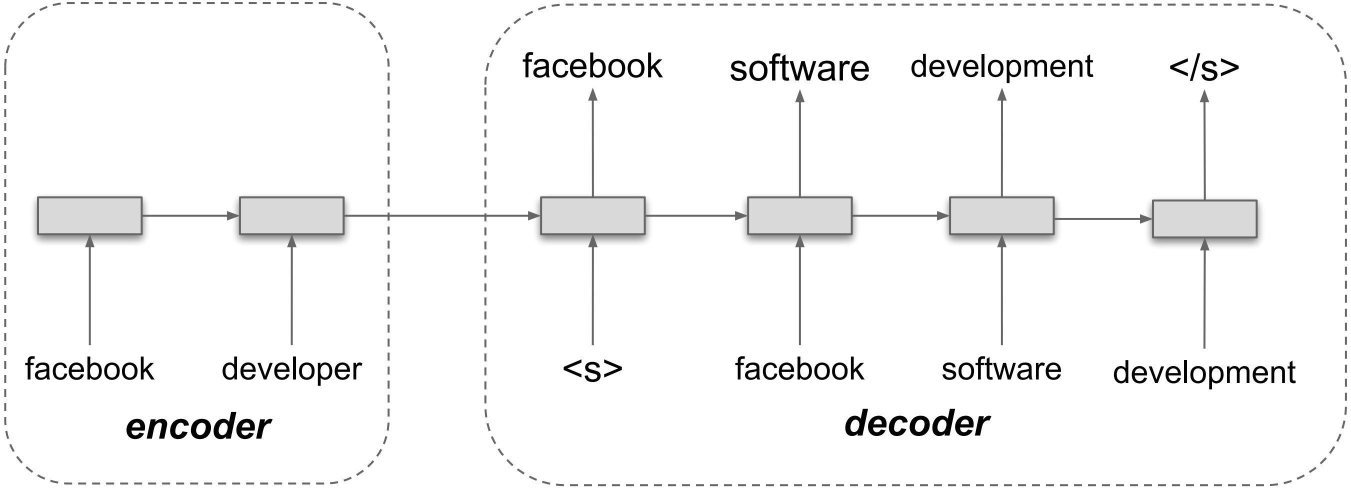
conversational seq2seq

GAN for seq2seq

Query Reformulation as a Translation Task

(Sordoni et al. 2015, He et al., 2016)

- Sequence-to-sequence modeling



- Directly modeling the words in a query

Conversational Query Reformulation

(Ren et al, 2018)

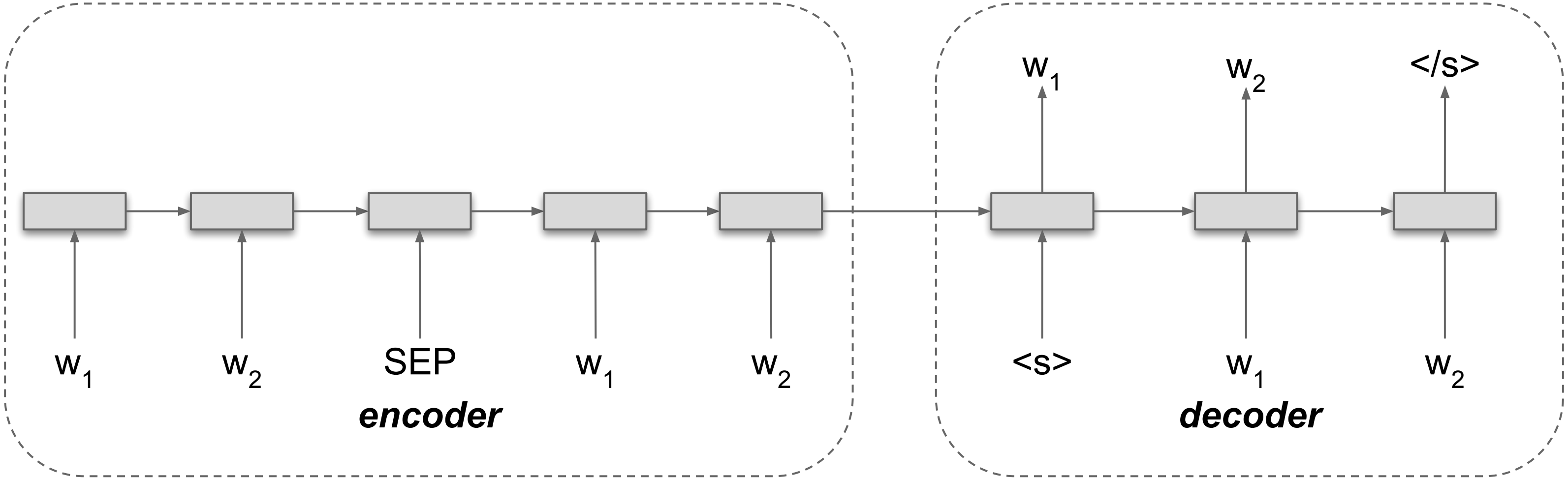
- Conversational queries
- Goal: summarize q1 & q2 into q3

first query (q1)	**second query (q2)**	**summarized query (q3)**
when was California founded?	who is **its** governor?	who is California's governor?
California	population in 1990	population of California in 1990
how tall is kobe bryant?	**what about** Lebron James?	how tall is Lebron James?
when was the last summer Olympics?	and the winter one?	when was the last winter Olympics?

Conversational Query Reformulation

(Ren et al, 2018)

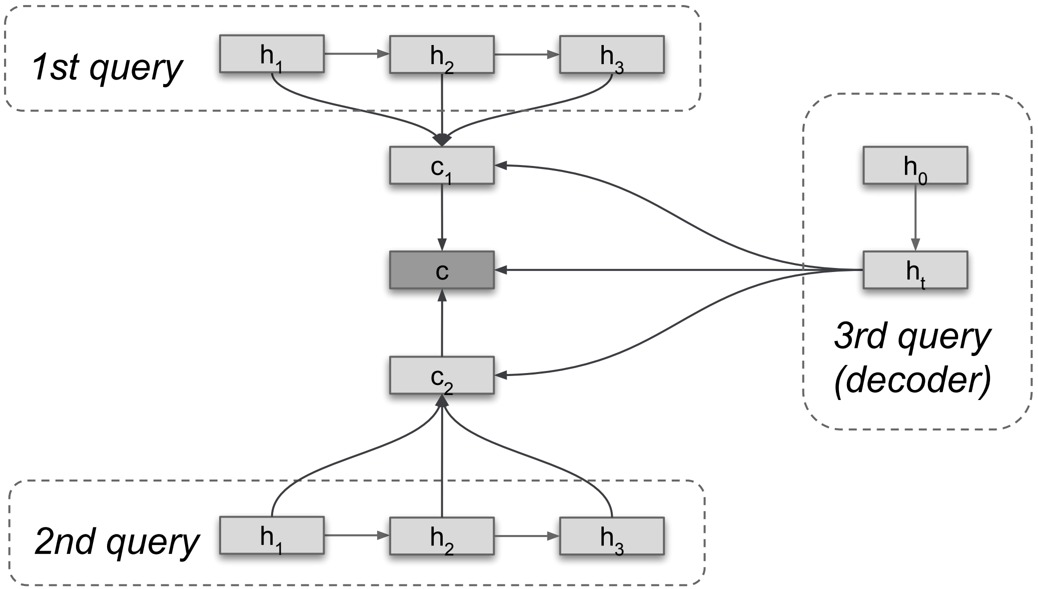
- 1st: concatenated seq2seq



Conversational Query Reformulation

(Ren et al, 2018)

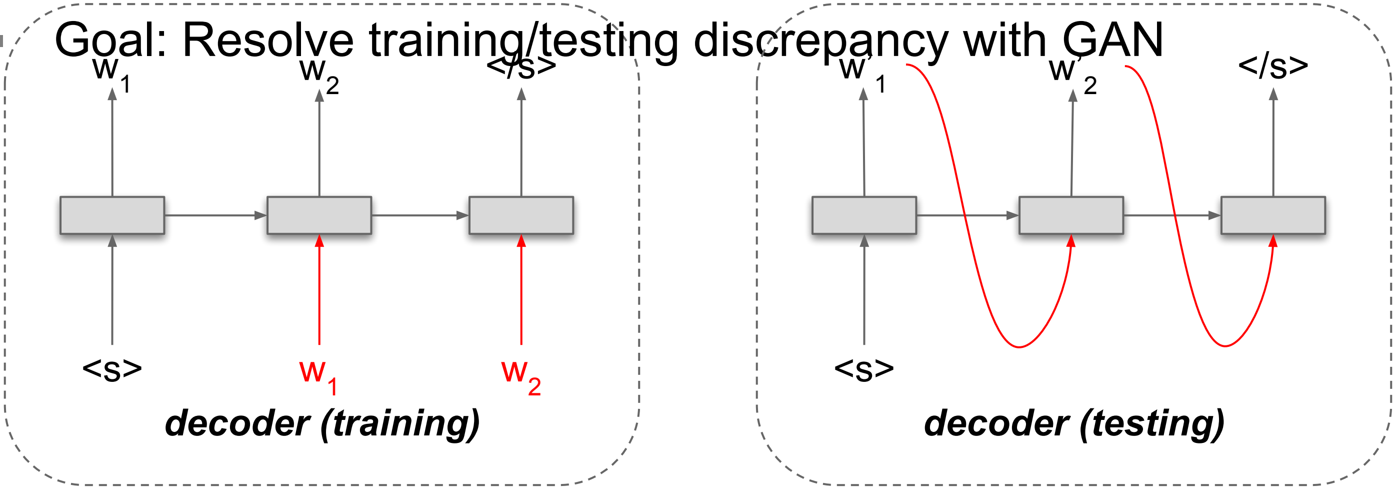
- 2nd: Attention over attention



GAN for seq2seq

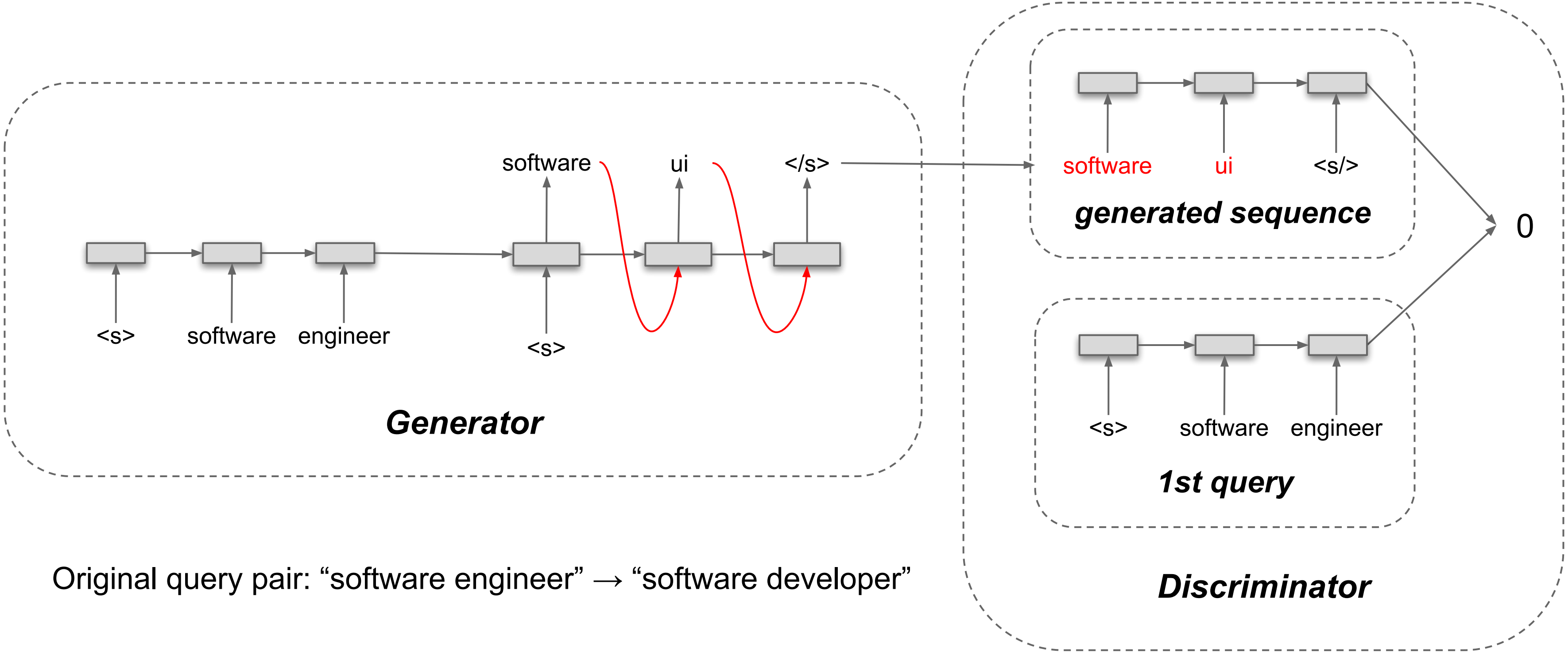
(Lee et al, 2018)

- Motivation: Training/testing discrepancy in decoder of seq2seq
  - Training: inputs are the groundtruth words
  - Testing: inputs are the previous predicted words
- Goal: Resolve training/testing discrepancy with GAN



GAN for seq2seq

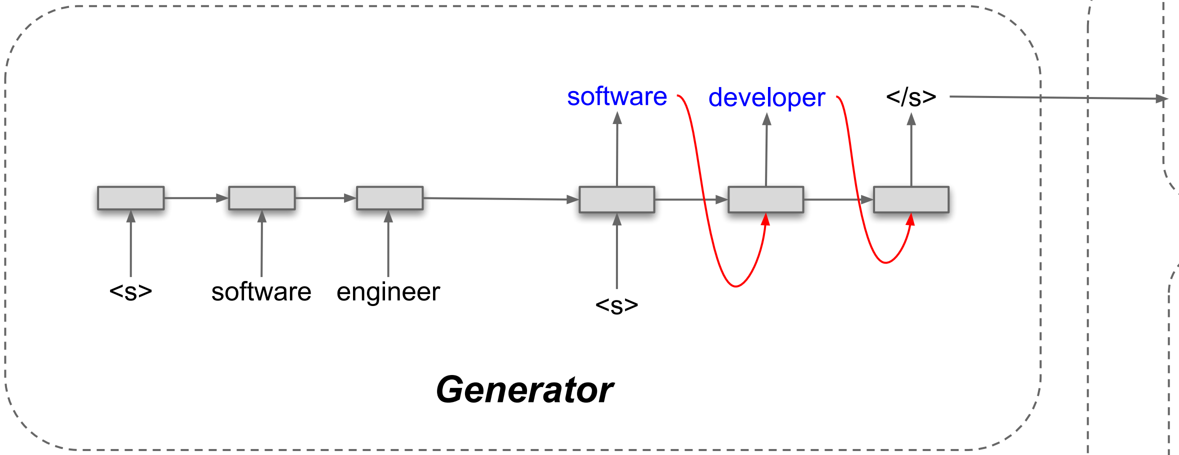
(Lee et al, 2018)

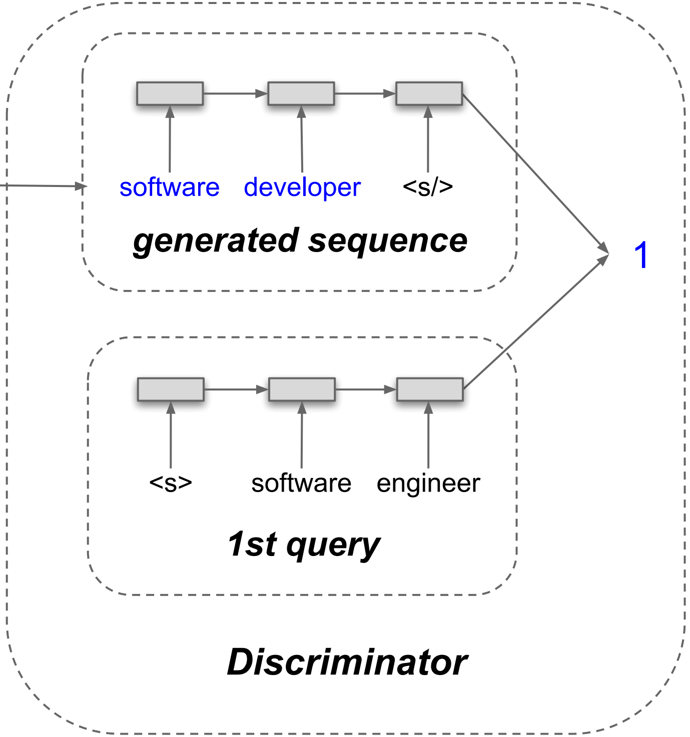


Original query pair: “software engineer” → “software developer”

GAN for seq2seq

(Lee et al, 2018)





Original query pair: “software engineer” → “software developer”

Query Reformulation: Summary

- seq2seq framework:
  - Directly modeling the words
  - Very flexible to incorporate session information
  - Achieves great performance (no character modeling)

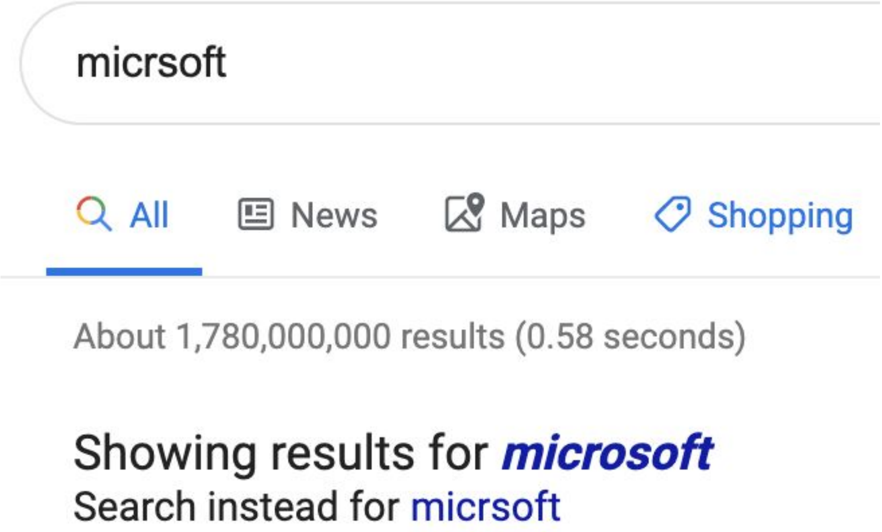
Reference

- Reda, Azarias, Yubin Park, Mitul Tiwari, Christian Posse, and Sam Shah. "Metaphor: a system for related search recommendations." In Proceedings of the 21st ACM international conference on Information and knowledge management, pp. 664-673. ACM, 2012.
- He, Yunlong, Jiliang Tang, Hua Ouyang, Changsung Kang, Dawei Yin, and Yi Chang. "Learning to rewrite queries." In Proceedings of the 25th ACM International on Conference on Information and Knowledge Management, pp. 1443-1452. ACM, 2016.
- Ren, Gary, Xiaochuan Ni, Manish Malik, and Qifa Ke. "Conversational query understanding using sequence to sequence modeling." In Proceedings of the 2018 World Wide Web Conference, pp. 1715-1724. International World Wide Web Conferences Steering Committee, 2018.
- Lee, Mu-Chu, Bin Gao, and Ruofei Zhang. "Rare query expansion through generative adversarial networks in search advertising." In Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining, pp. 500-508. ACM, 2018.
- Cao, Huanhuan, Daxin Jiang, Jian Pei, Qi He, Zhen Liao, Enhong Chen, and Hang Li. "Context-aware query suggestion by mining click-through and session data." In Proceedings of the 14th ACM SIGKDD international conference on Knowledge discovery and data mining, pp. 875-883. ACM, 2008.
- Sordoni, Alessandro, Yoshua Bengio, Hossein Vahabi, Christina Lioma, Jakob Grue Simonsen, and Jian-Yun Nie. "A hierarchical recurrent encoder-decoder for generative context-aware query suggestion." In Proceedings of the 24th ACM International on Conference on Information and Knowledge Management, pp. 553-562. ACM, 2015.
- Mei, Qiaozhu, Dengyong Zhou, and Kenneth Church. "Query suggestion using hitting time." In Proceedings of the 17th ACM conference on Information and knowledge management, pp. 469-478. ACM, 2008.

Deep NLP in Search and Recommender Systems

- Language Understanding
  - Entity Tagging: word level prediction
  - Entity Disambiguation: knowledge base entity prediction
  - Intent Classification: sentence level prediction
  - Sentiment Analysis and Opinion Mining
- Document Retrieval and Ranking
  - Efficient Candidate Retrieval
  - Deep Ranking Models
- Language Generation for Assistance
  - Auto Completion
  - Query Reformulation
  - **Spell Correction**
  - Conversational Recommendation

Spell Correction



Spell Correction

- Why spell correction:
  - Reduce the no results
- Challenges:
  - Many rare words (people/company names) look like misspelled words
  - Modeling characters and words at the same time

query	**similar query**	**has error?**
tumblr	*tumble*	No
tumblw	*tumble*	Yes
galaxy s10e	*galaxy s10*	No
galaxy s10d	*galaxy s10*	Yes

Agenda

Candidate Generation for a word → Candidate generation for a query → Candidate ranking

Traditional methods

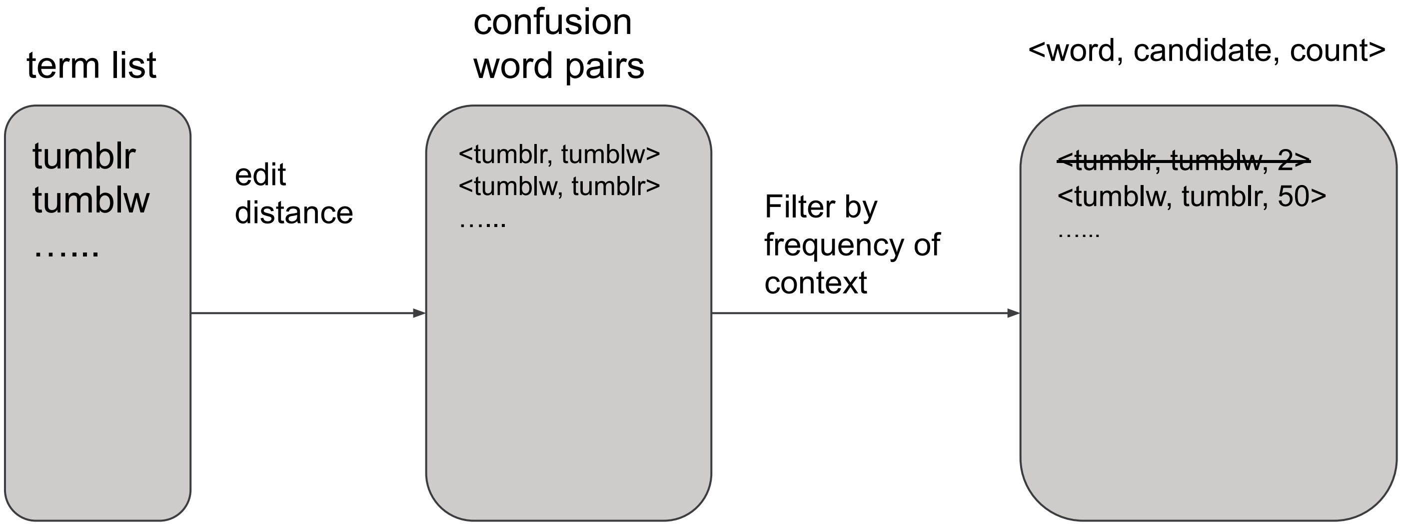
Candidate Generation for a Word

(Whitelaw et al 2009)

- Goal:
  - given “tumblw”, suggest “tumblr”, “tumble”...
  - given “tumblr”, suggest “tumble”, but not “tumblw”
- Key challenge: what is a legit word? (“tumblr”👍 “tumblw”👎)
  - Coverage of a dictionary is not enough
- Solution: use statistics in web noisy data
  - Correct words appear more frequent than incorrect words

Candidate Generation for a Word

(Whitelaw et al 2009)



- For the context “social media X”:
  - freq(“social media tumblw”) < freq(“social media tumblr”)

Candidate Generation for a Query

(Chen et al 2007)

earr pian kids

ear, era, ... *pain, pan, ...* *kids, kid, ...*

ear pain kids,
era pan kid,
ear pin kid,
…….

- Problem: query candidate size grows exponentially with # of words
- Solution: prune with language model
  - ***ear*** pian kids: 0.8
  - ~~***era*** pian kids: 0.1~~
  - earr ***pain*** kids: 0.9
  - earr ***pan*** kids: 0.7
  - ~~earr ***pien*** kids: 0.2~~
  - earr pian kid: 0.8

Candidate Ranking

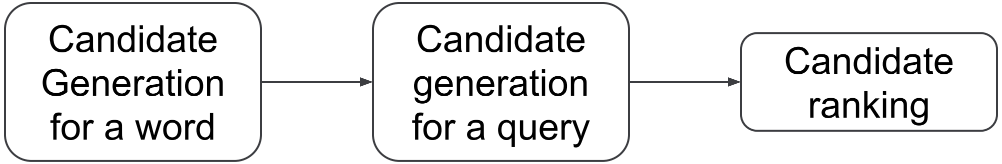
(Li et al 2006, Chen et al 2007)

$$\underbrace{}_{}\text{candidate}\quad\text{query}$$

$$\text{argmax}_c P(c|q) = \text{argmax}_c P(q|c)P(c)$$

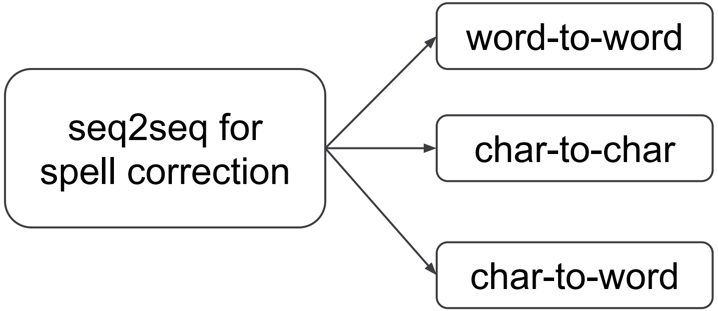
Feature Types	Examples
similarity $P(q\|c)$	Edit distance
	Frequency of user reformulation
	……
Likelihood $P(c)$	Language model score of the candidate
	Frequency of candidate terms appearing in the page titles
	……

Agenda



Traditional methods

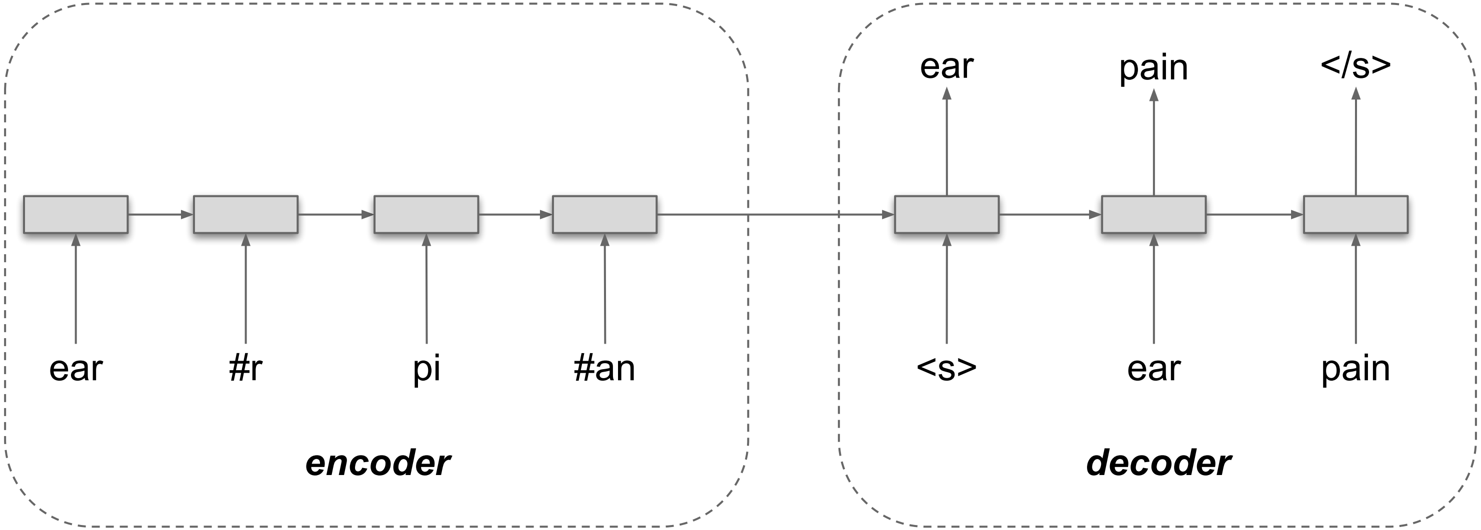
Deep NLP methods



Seq2seq for Spell Correction

(Ghosh and Kristensson, 2017, Zhou et al., 2017)

- From subwords to subwords



Seq2seq for Spell Correction

(Ghosh and Kristensson, 2017, Zhou et al., 2017)

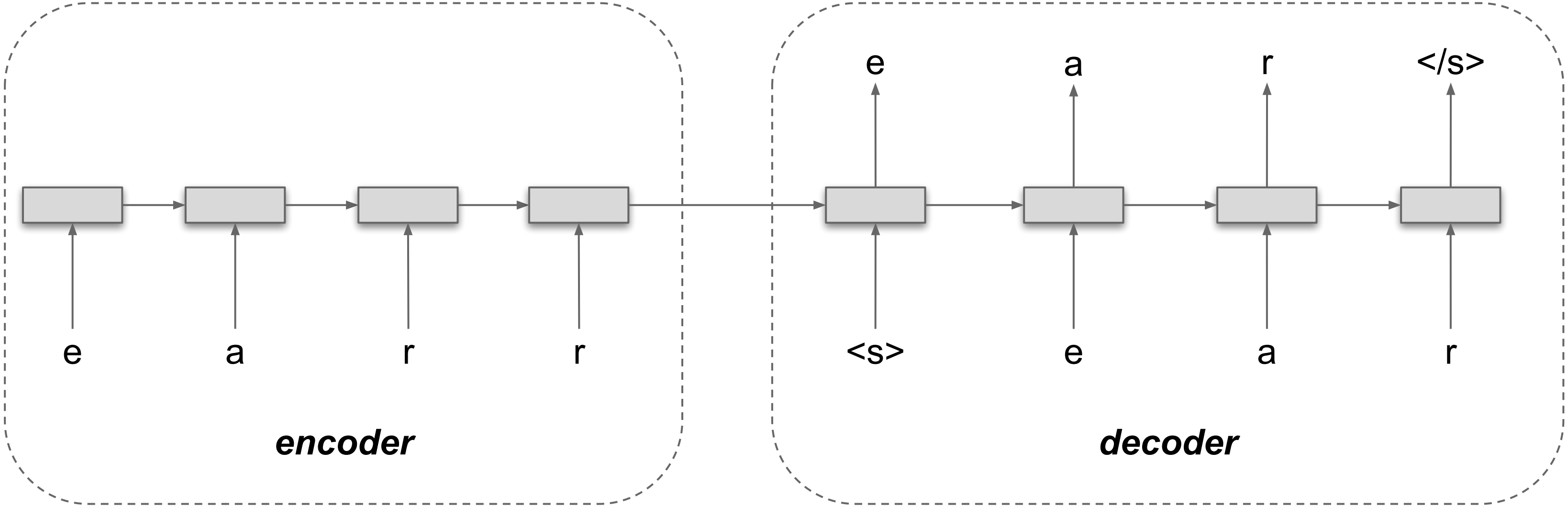
- From subwords to subwords
- Issue: subword is designed for texts without errors

	Normal texts	**Spell errors**
Example	*hunter → hunt #er*	*hunetr → hu #net #r*
Subword semantics	*relevant*	*irrelevant*

Seq2seq for Spell Correction

(Ghosh and Kristensson, 2017, Zhou et al., 2017)

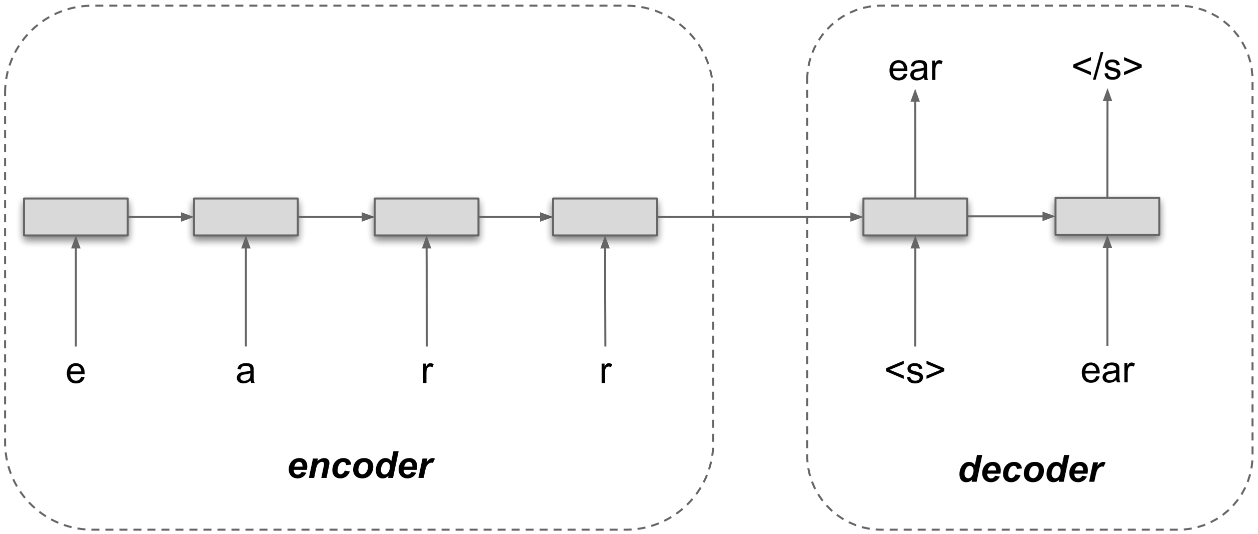
- From characters to characters
- Issue: on the decoder, no word information
  - Might produce words with wrong spelling



Seq2seq for Spell Correction

(Ghosh and Kristensson, 2017, Zhou et al., 2017)

- From characters to words
  - Most popular structure
  - Can leverage pretrained language model



Reference

- Hasan, Saša, Carmen Heger, and Saab Mansour. "Spelling correction of user search queries through statistical machine translation." In Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing, pp. 451-460. 2015.
- Yingbo Zhou, Utkarsh Porwal, Roberto Konow. "Spelling Correction as a Foreign Language." arXiv. 2017.
- Ghosh, Shaona, and Per Ola Kristensson. "Neural networks for text correction and completion in keyboard decoding." arXiv preprint arXiv:1709.06429 (2017).
- Chen, Qing, Mu Li, and Ming Zhou. "Improving query spelling correction using web search results." In Proceedings of the 2007 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning (EMNLP-CoNLL), pp. 181-189. 2007.
- Li, Mu, Yang Zhang, Muhua Zhu, and Ming Zhou. "Exploring distributional similarity based models for query spelling correction." In Proceedings of the 21st International Conference on Computational Linguistics and the 44th annual meeting of the Association for Computational Linguistics, pp. 1025-1032. Association for Computational Linguistics, 2006.
- Whitelaw, Casey, Ben Hutchinson, Grace Y. Chung, and Gerard Ellis. "Using the web for language independent spellchecking and autocorrection." In Proceedings of the 2009 Conference on Empirical Methods in Natural Language Processing: Volume 2-Volume 2, pp. 890-899. Association for Computational Linguistics, 2009.

Deep NLP in Search and Recommender Systems

- Language Understanding
  - Entity Tagging: word level prediction
  - Entity Disambiguation: knowledge base entity prediction
  - Intent Classification: sentence level prediction
  - Sentiment Analysis and Opinion Mining
- Document Retrieval and Ranking
  - Efficient Candidate Retrieval
  - Deep Ranking Models
- Language Generation for Assistance
  - Auto Completion
  - Query Reformulation
  - Spell Correction
  - **Conversational Recommendation**

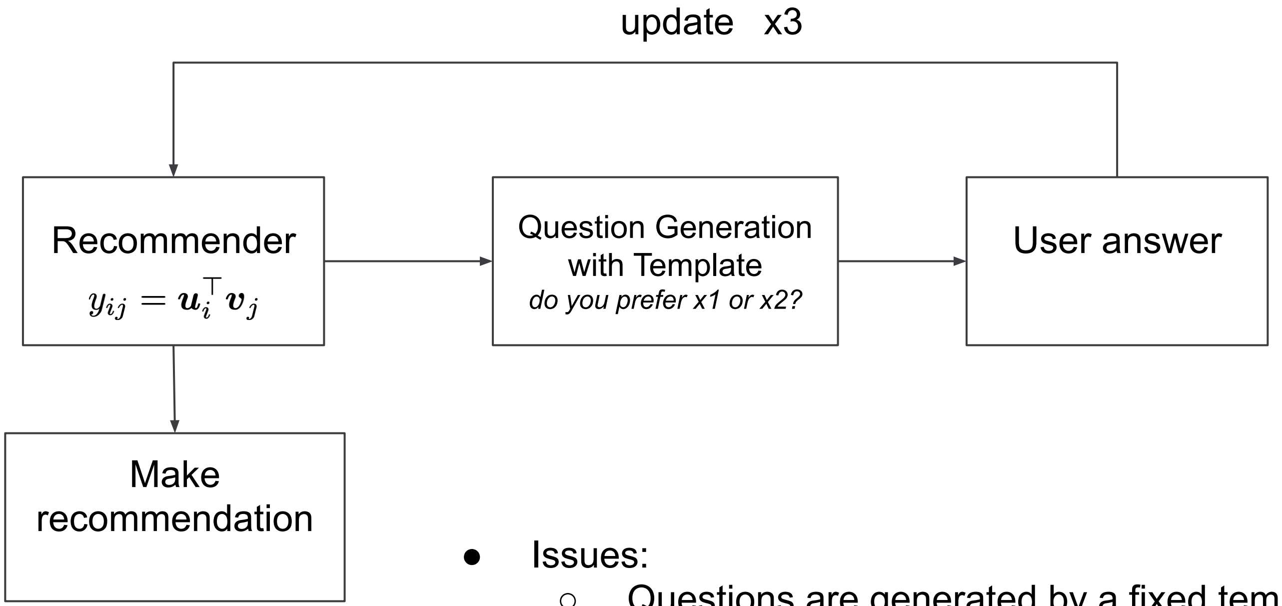
Conversational recommender systems

- Motivation:
  - Actively gather more information or resolve ambiguity
  - Better user experience

- Challenges:
  - What to ask to the user
  - Ask vs Rec

Traditional Methods: Collaborative Filtering

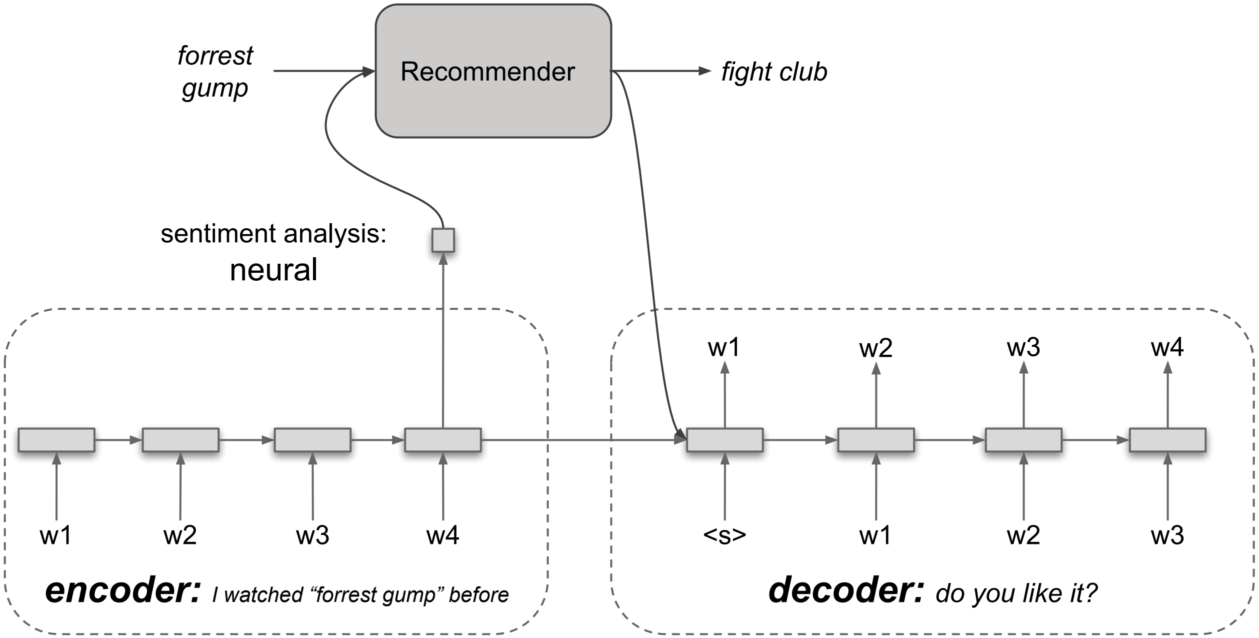
(Christakopoulou et al., 2016)



- Issues:
  - Questions are generated by a fixed template
  - No modeling of ask vs rec

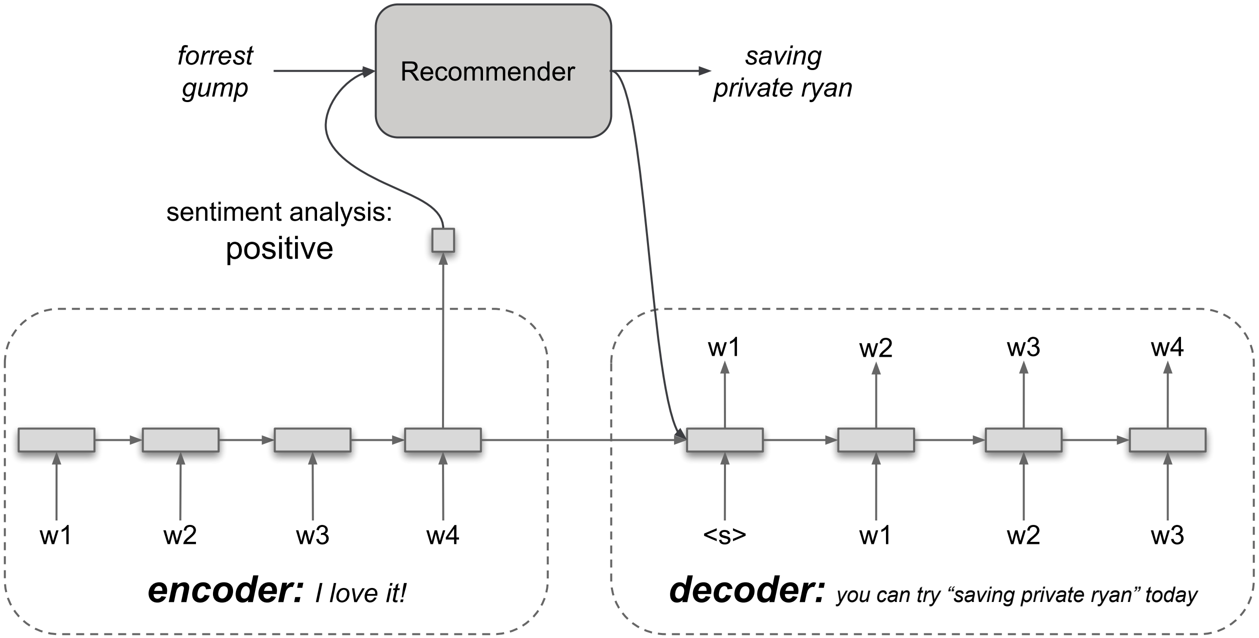
Seq2seq with Recommender Systems

(Li et al., 2018)



Seq2seq with Recommender Systems

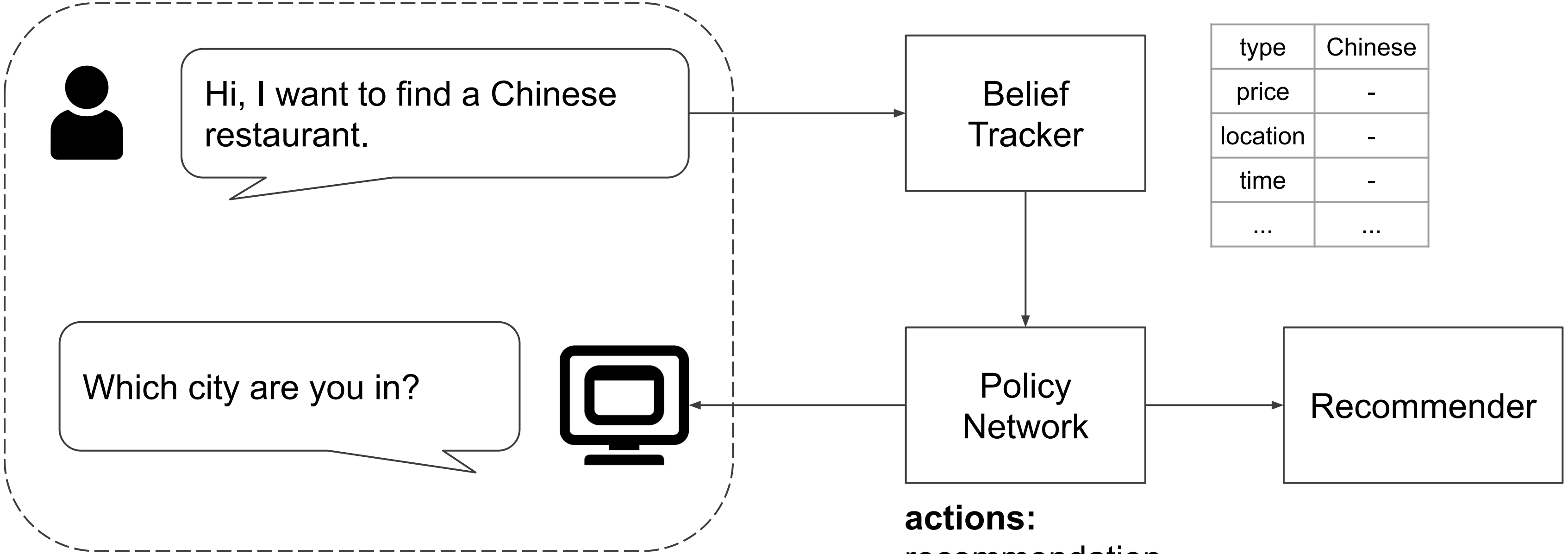
(Li et al., 2018)



- Issue: only sentiment is encoded;
  not applicable for facets, such as directed-by, release date

Recommendation with Belief Tracker

(Dhingra et al., 2016, Sun & Zhang, 2018)

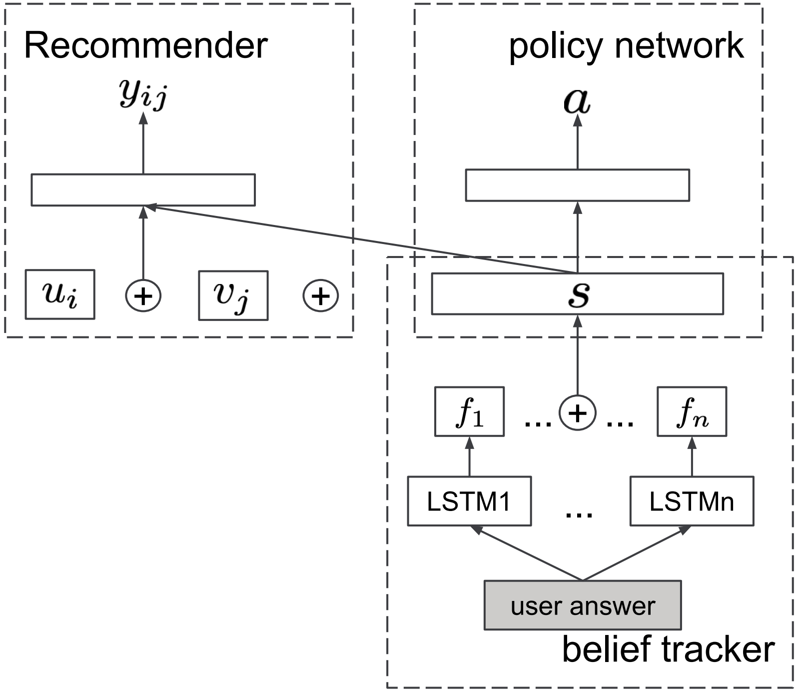


type	Chinese
price	-
location	-
time	-
...	...

actions:
recommendation vs.
ask_for_price vs.
ask_for_location...

Recommendation with Belief Tracker

(Dhingra et al., 2016, Sun & Zhang, 2018)



- Reinforcement learning to consider future actions

rec
location → rec
food_type → location → rec

$r_0 + \gamma r_1 + \gamma^2 r_2 + \ldots$

Reference

- Sun, Yueming, and Yi Zhang. "Conversational recommender system." In The 41st International ACM SIGIR Conference on Research & Development in Information Retrieval, pp. 235-244. ACM, 2018.
- Li, Raymond, Samira Ebrahimi Kahou, Hannes Schulz, Vincent Michalski, Laurent Charlin, and Chris Pal. "Towards deep conversational recommendations." In Advances in Neural Information Processing Systems, pp. 9725-9735. 2018.
- Christakopoulou, Konstantina, Filip Radlinski, and Katja Hofmann. "Towards conversational recommender systems." In Proceedings of the 22nd ACM SIGKDD international conference on knowledge discovery and data mining, pp. 815-824. ACM, 2016.
- Dhingra, Bhuwan, Lihong Li, Xiujun Li, Jianfeng Gao, Yun-Nung Chen, Faisal Ahmed, and Li Deng. "Towards end-to-end reinforcement learning of dialogue agents for information access." arXiv preprint arXiv:1609.00777 (2016).
- Christakopoulou, Konstantina, Filip Radlinski, and Katja Hofmann. "Towards conversational recommender systems." In Proceedings of the 22nd ACM SIGKDD international conference on knowledge discovery and data mining, pp. 815-824. ACM, 2016.

Language Generation for Search: Summary

	Traditional methods	**deep NLP methods**
Candidate generation	Rule based	End-to-end solution Language modeling
Candidate ranking	Few features	
Latency	Low	High

Agenda

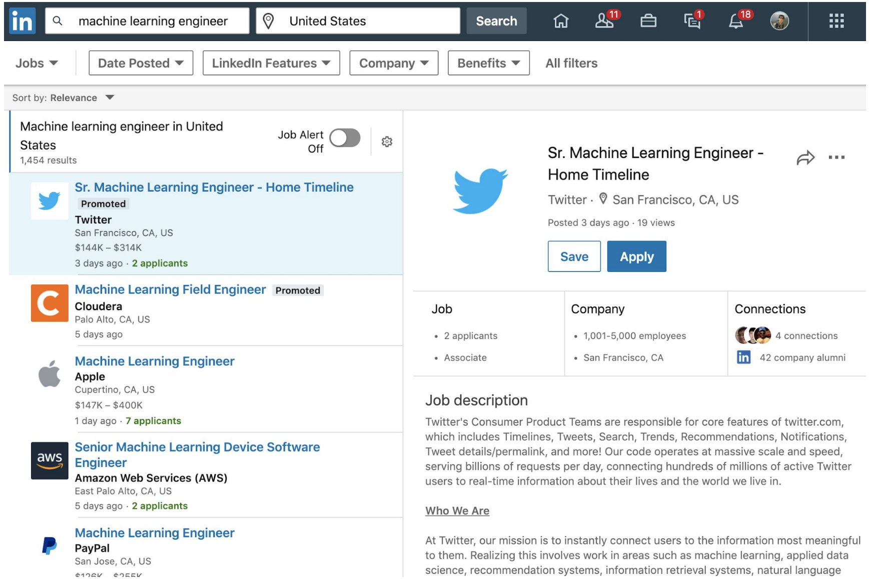
1. Introduction
2. Deep Learning for Natural Language Processing
3. Deep NLP in Search and Recommender Systems
4. **Real World Examples**

Deep NLP in Search and Recommender Systems - Real World Examples at LinkedIn

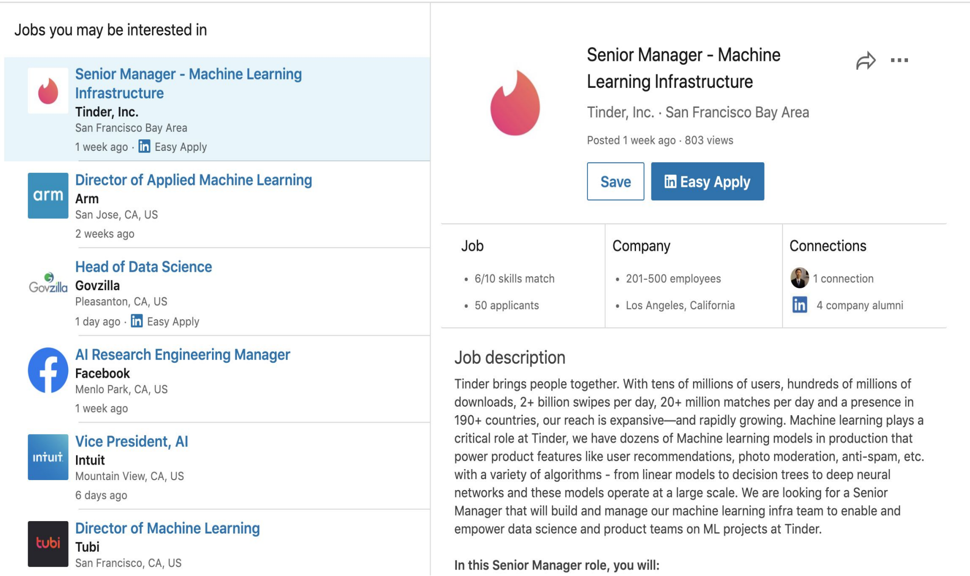
Huiji Gao

LinkedIn Search and Recommender Systems

Job Search



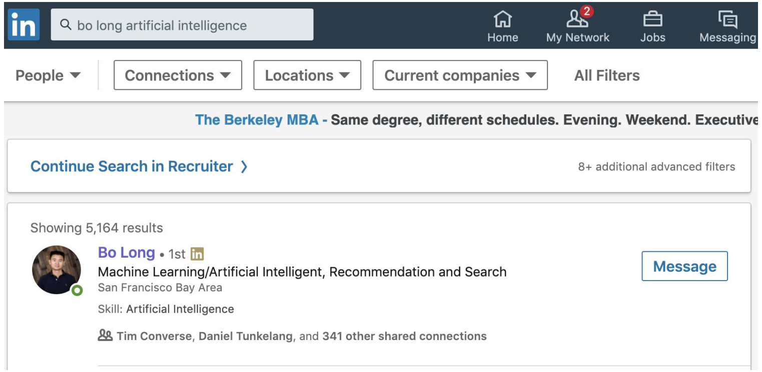
Job Recommendation



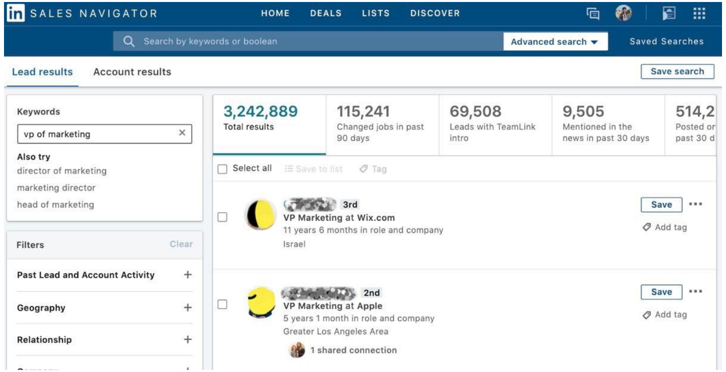
Rank job posts for LinkedIn member to help the user to find jobs that he/she would like to apply for (then, click the “apply” button)

LinkedIn Search and Recommender Systems

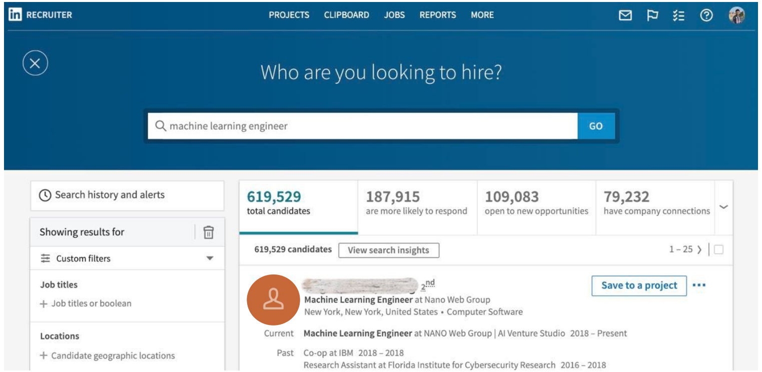
People Search



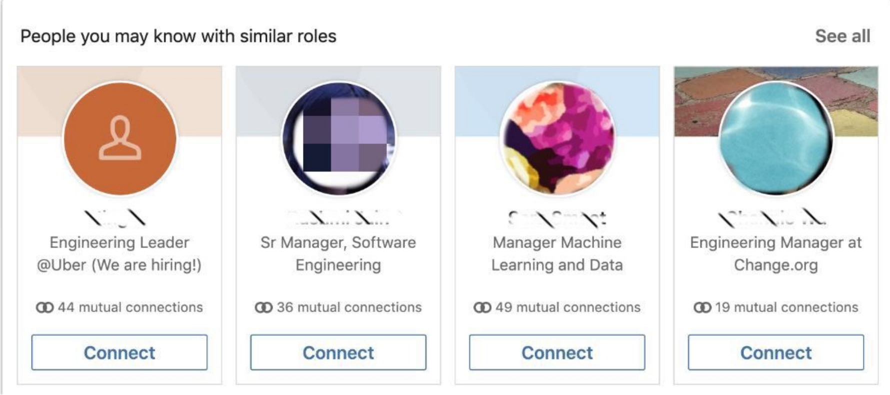
Lead Search



Recruiter Search



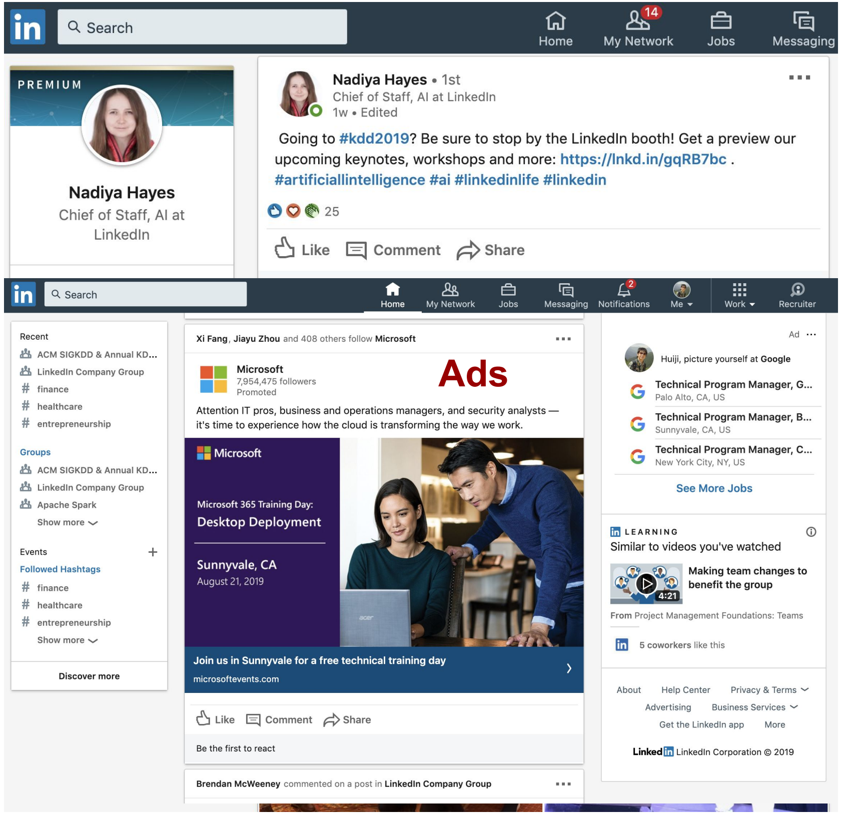
People Recommendation



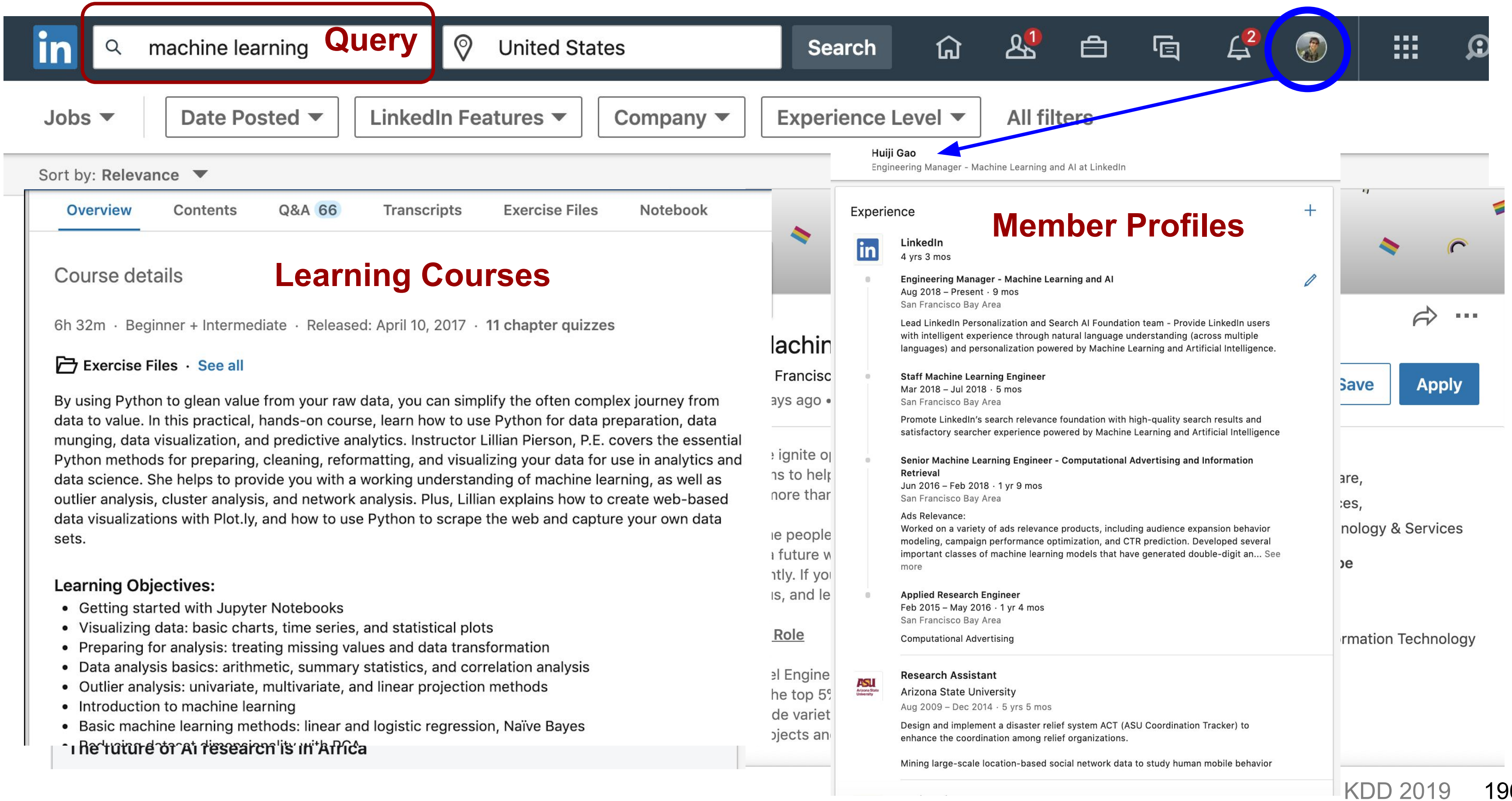
LinkedIn Search and Recommender Systems

Content Search

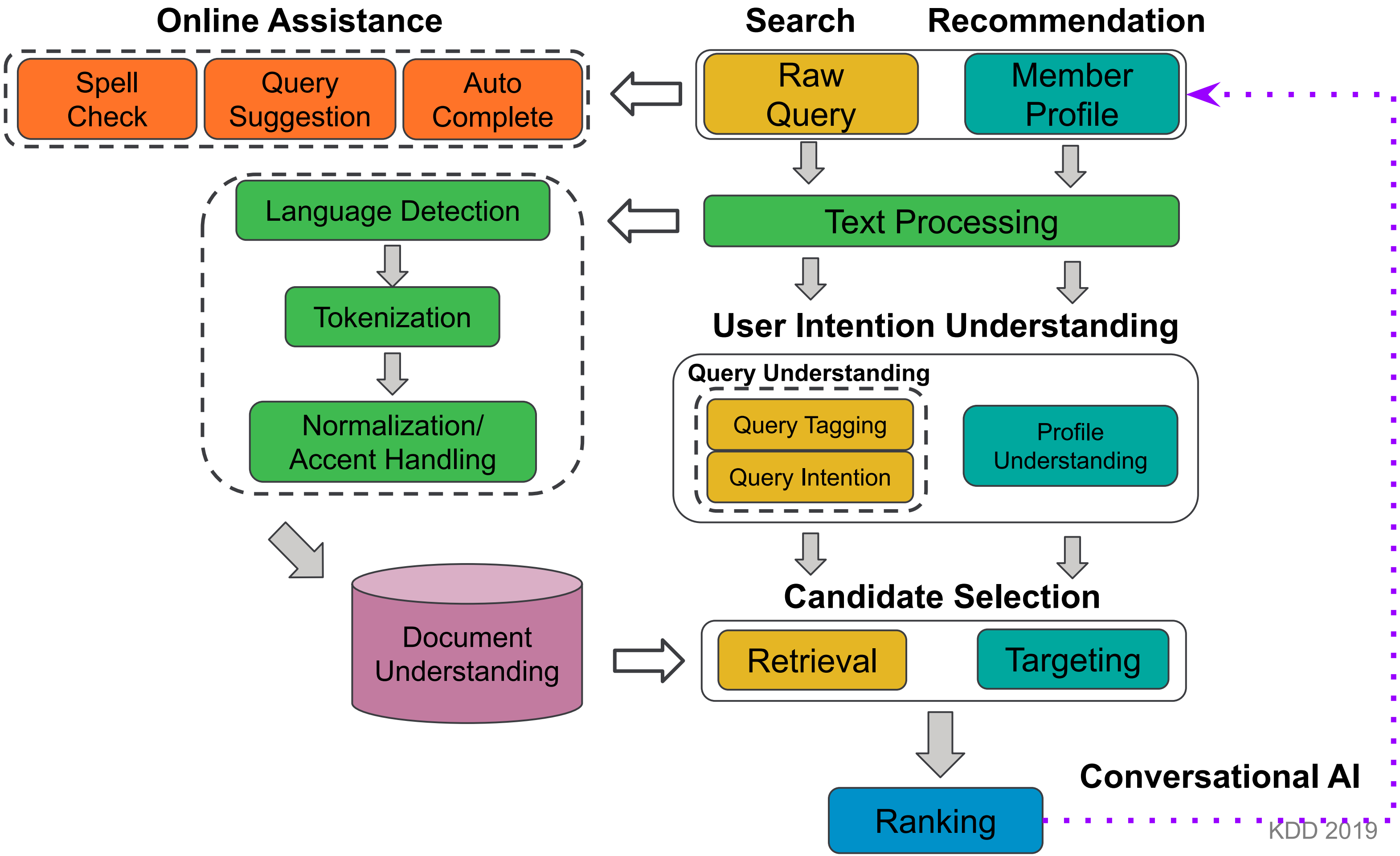
Content Recommendation



Natural Language Data at LinkedIn



LinkedIn Search and Recommendation Ecosystem



NLP in LinkedIn: Challenges

- **Data Ambiguity**
  - Short Query Text
    - "abc"

ABC News? ABC Stores?

  - No Strict Syntax
    - "bing search engineer"

"Bing Search, Engineer" **"Bing, Search Engineer"**

  - Personalization
    - "looking for new jobs"

Job Seeker looks for jobs

Recruiter looks for candidates

- **Deep Semantics**
  - Representations of documents w.r.t. member intent
    - "developer proficient at cloud computing" -> "Azure, AWS ..."
    - "engineering openings" -> Job Posts

Deep NLP in LinkedIn Search & Recommendation: Challenges

- **Product Oriented Model Design**
  - Design deep NLP algorithms for specific search & recommendation components
  - Consider business rules, post filters, results blender, user experience, etc

- **Model Debugging**
  - Understand model for online performance investigation

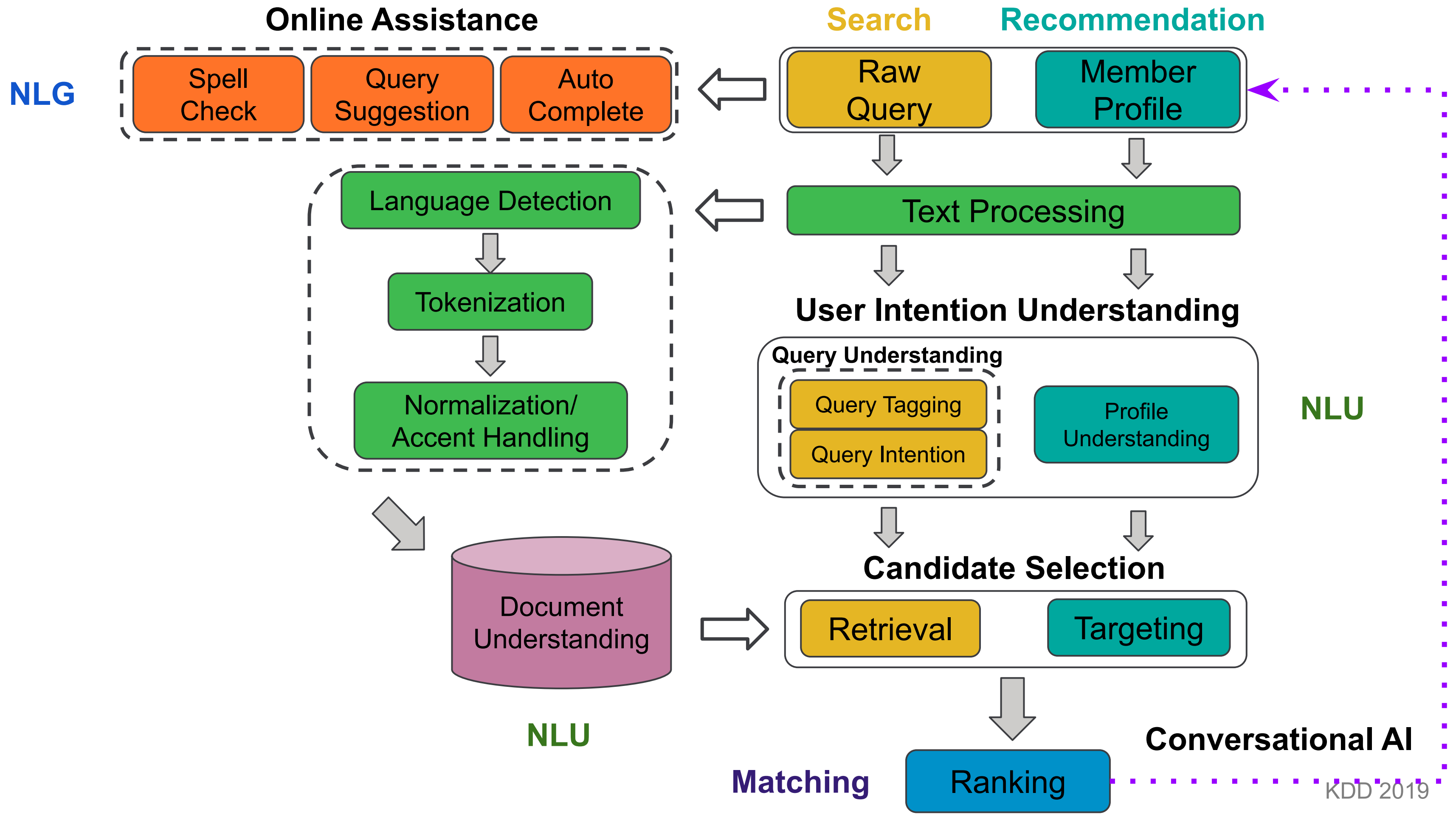
- **Online Latency**
  - Serving deep NLP models with product latency restriction

Applying Deep NLP in LinkedIn Search & Recommendation

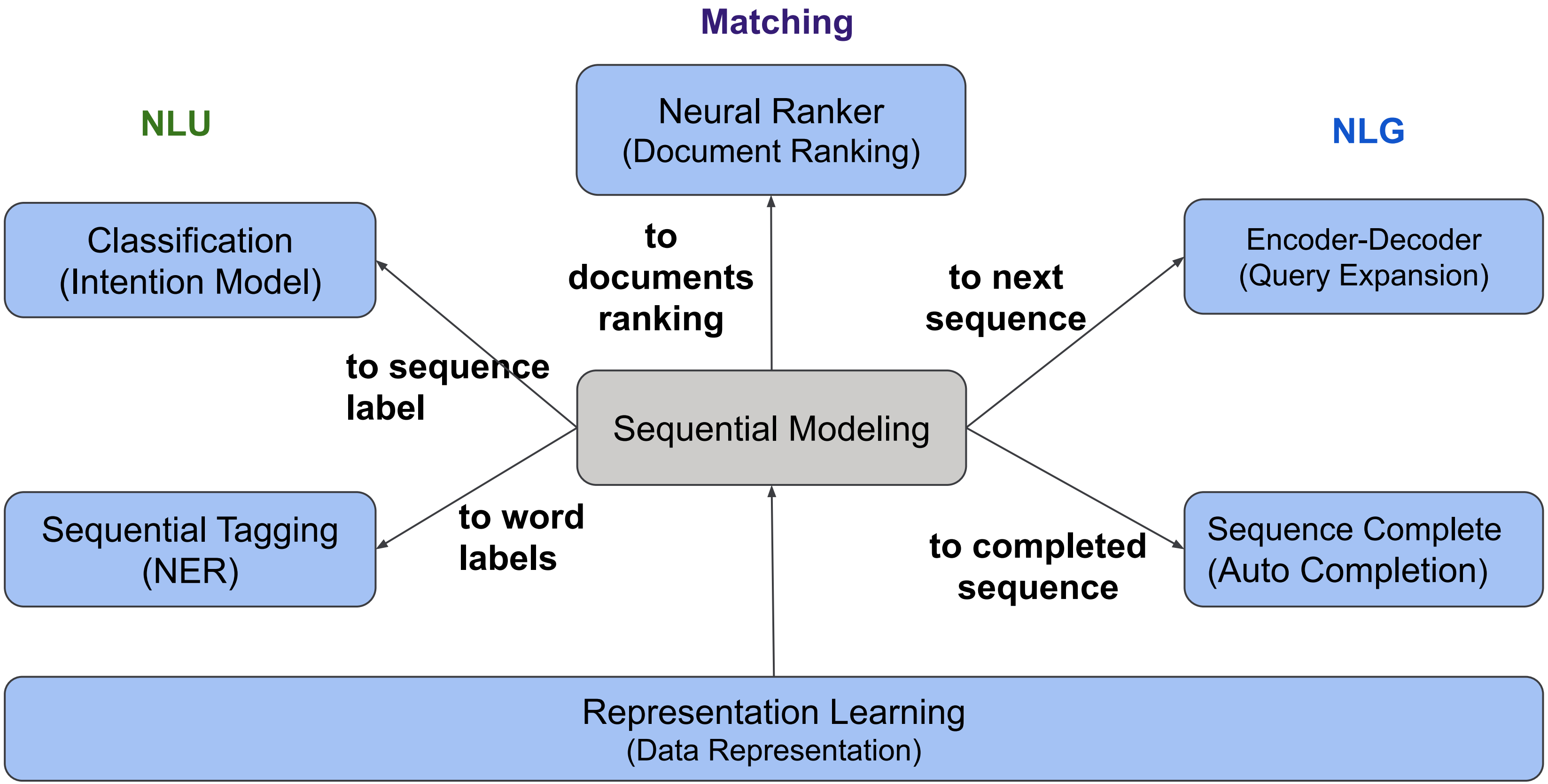
- **Feature Driven**
  - Representation Learning
    - Using features generated from deep learning models
      e.g., word embedding

- **Model Driven**
  - Power product features directly with deep learning models
    - CNN/LSTM/Seq2seq/GAN/BERT based deep NLP models

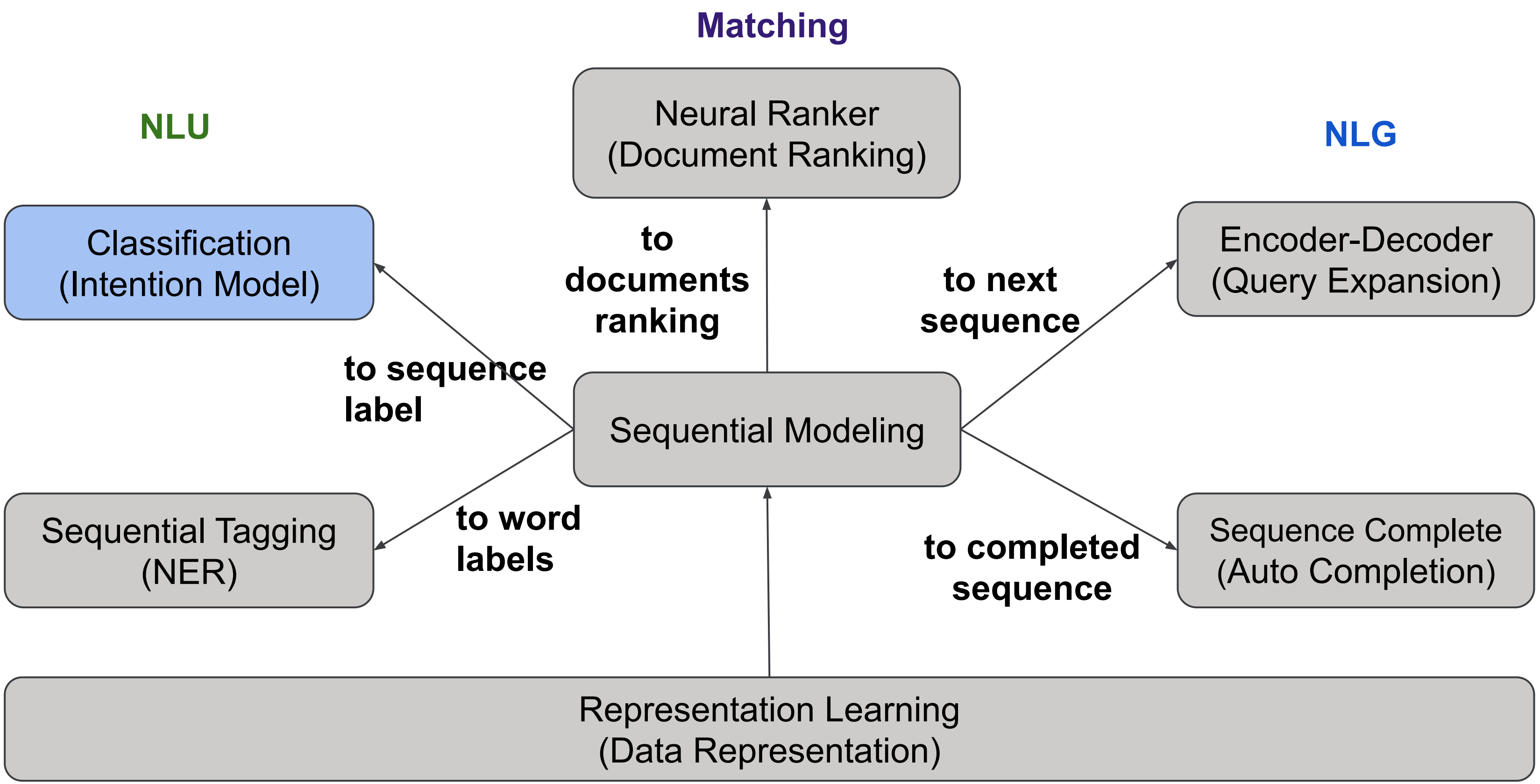
LinkedIn Search and Recommendation Ecosystem



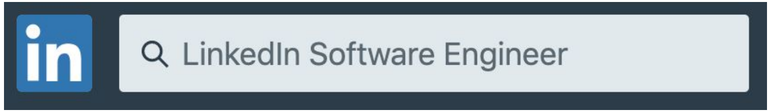
Deep Learning for Natural Language Processing



Deep Learning for Natural Language Processing



Query Intention Model: Goal

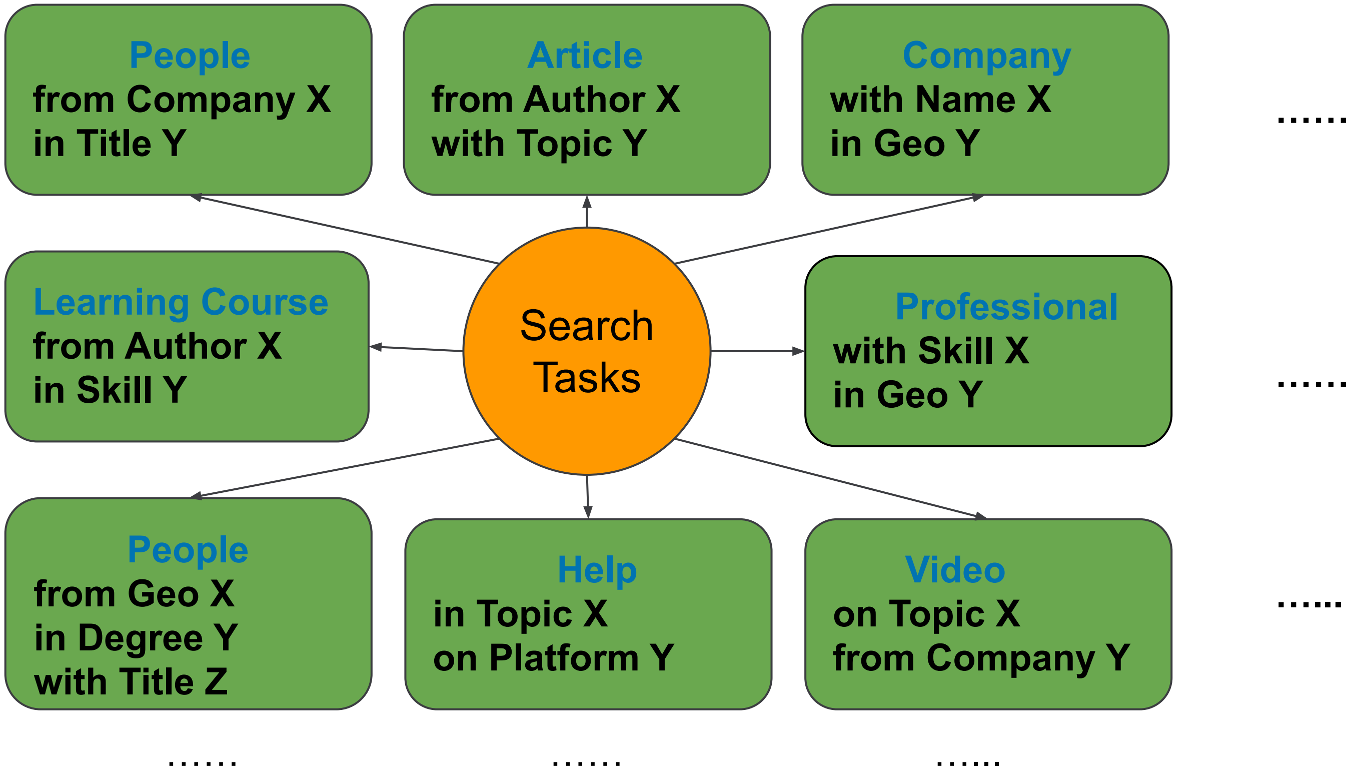


- **Output of Query Intention Model**
  - Search Vertical Prediction
    - **People, Job Posts, Articles ...**
  - Properties Associated with the Vertical
    - **Extracted** from raw query **OR** **Inferred** from other information

The searcher is looking for:

0.99	0.06	0.03	
People from "**LinkedIn**" as a "**Software Engineer**"	**Job Post** from "**LinkedIn**" on "**Software Engineer**" position	**Article** from "**LinkedIn**" on "**Software Engineer**" topic	......

Query Intention Model: Task Oriented Intention Prediction



Query Intention: Member Search Footprint

Bridging the Gap between Search and Recommendation

Query Intention Model: Goal

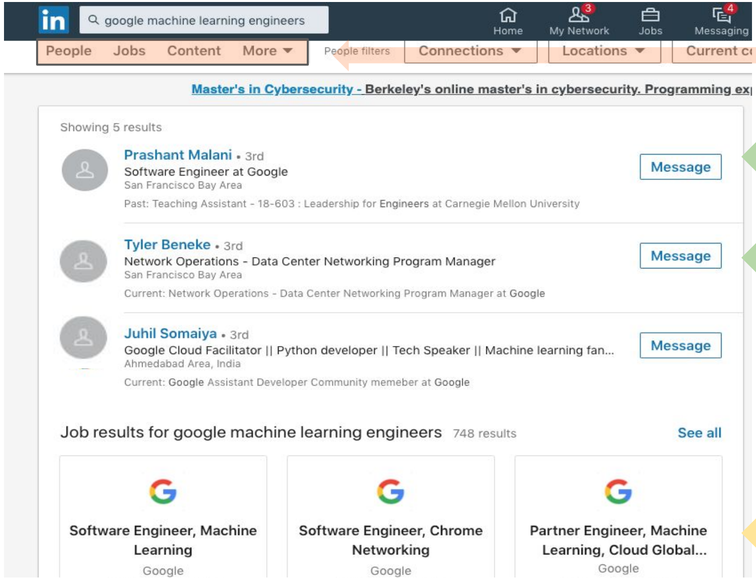
Query: LinkedIn Software Engineer

- **Output of Query Intention Model**
  - Vertical Prediction
    - **People, Job Posts, Articles ...**
  - Properties Associated with the Vertical
    - **Extracted** from raw query **OR Inferred** from other information

The searcher is looking for:

0.99	0.06	0.03	
People from "**LinkedIn**" as a "**Software Engineer**"	**Job Post** from "**LinkedIn**" on "**Software Engineer**" position	**Article** from "**LinkedIn**" on "**Software Engineer**" topic	......

Query Intention on Search Blending



Search verticals

Primary result

Primary result

Secondary cluster

Goal:

- To understand the vertical preference of user on LinkedIn search

Query Intention Model: Challenges

- Complicated Semantics

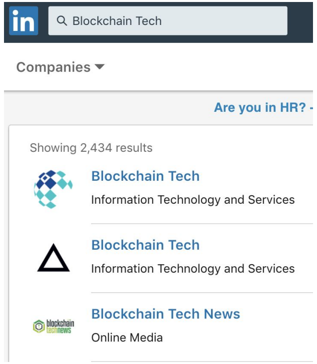
Job

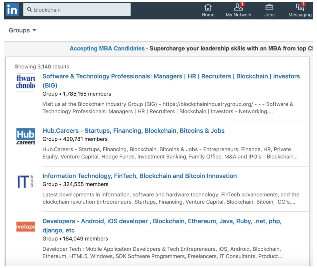
Job?

Job!

Query Intention Model: Challenges

- Personalization
  - **Query:** Blockchain Tech
    - **Job seeker** looks for Blockchain Technology job
    - **Company** named Blockchain Tech
    - **Learning course** on Blockchain Tech
    - **Content** on Blockchain Technology
    - **Video** about blockchain technology
    - **Recruiter** looks for candidates with Blockchain tech skill
    - ……





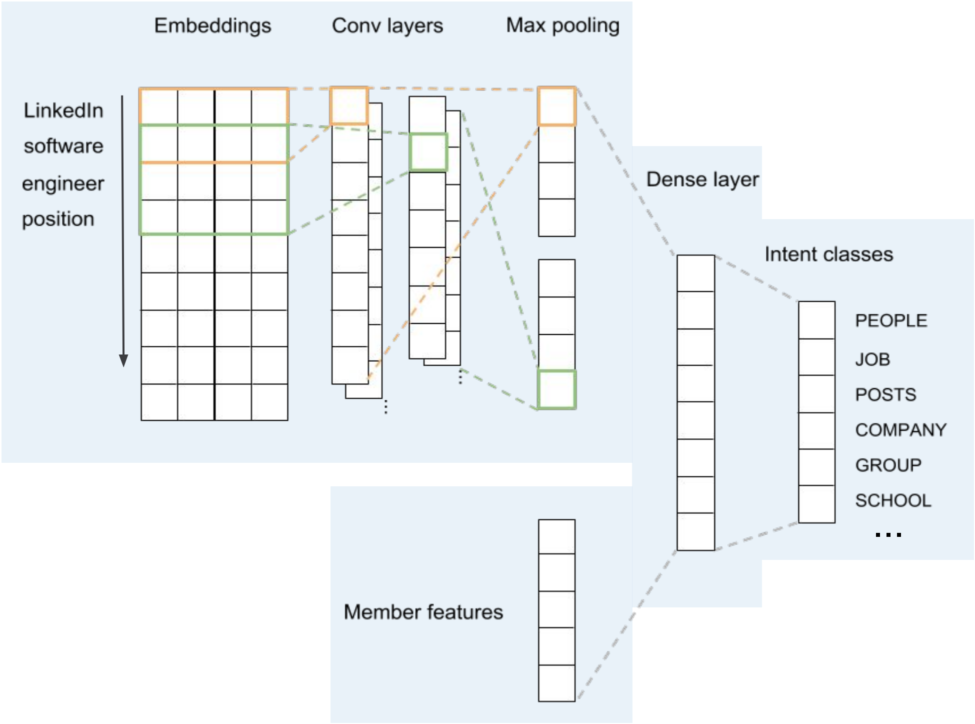
CNN Based Query Intention Model

CNN for Semantic Feature Extraction

- Word/query representations
- Generalization power
- Word n-gram patterns

Personalization

- Member-level Features



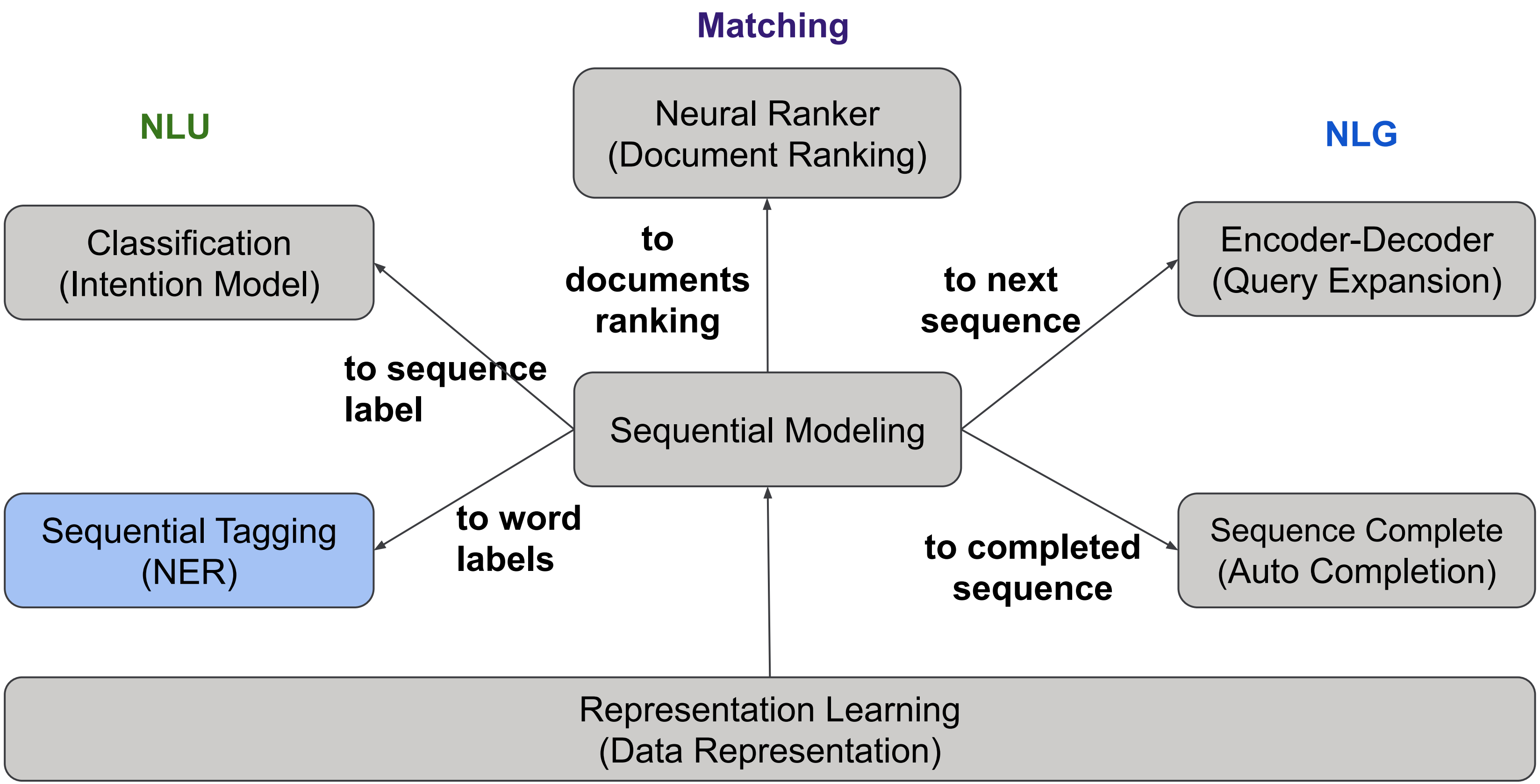
Query Intent - Experiment Results

- Offline Results

	Overall Accuracy	F1 on PEOPLE	F1 on JOB
Baseline (ML Model)	-	-	-
CNN	+2.9%	+11.9%	+1.7%

- Online Results
  - +0.65% Job CTR@1
  - +0.90% Overall Cluster CTR, +4.03% Cluster CTR via Entity Click

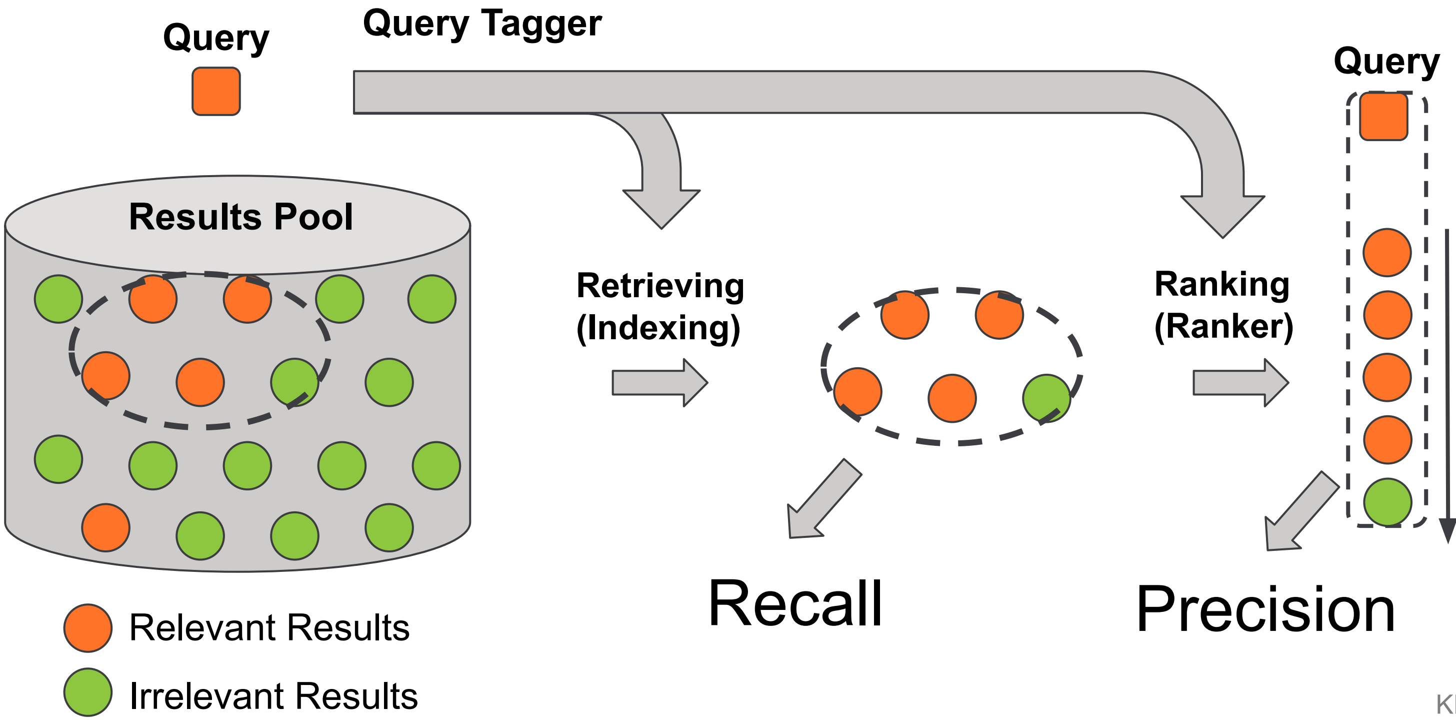
Deep Learning for Natural Language Processing



Entity Tagging at LinkedIn

- **Search Systems**

Query: Mike LinkedIn Software Engineer



Entity Tagging at LinkedIn

- **Understanding Queries & Documents with Entity Tagger**

Query: Mike LinkedIn Software Engineer

- **Query Tagger for Retrieval**

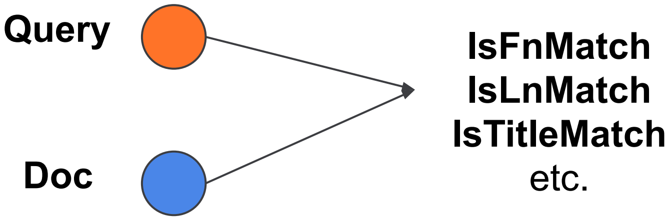
CN: company name
FN: first name
T: title

Mike	LinkedIn	Software	Engineer
FN	CN	T	T

Index

FN:{mike}	Doc1, **Doc2**, **Doc4**, ...
CN:{linkedin}	**Doc2**, **Doc4**, Doc5, Doc6, ...
T:{software}	**Doc2**, Doc3, **Doc4**, Doc7, ...
T:{engineer}	Doc1, **Doc2**, **Doc4**, Doc9, ...

- **Ranking Features**



Entity Tagging at LinkedIn

- **Recommender Systems**

Ads Recommendation

Who is your target audience? **Advertiser Targeting**

INCLUDE people who have **ANY** of the following attributes

Member Skills

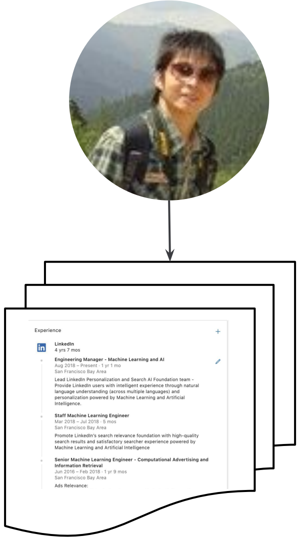
Customer Satisfaction ✕ Customer Experience Management ✕ Customer Experience ✕
Customer Acquisition ✕ Customer Retention ✕ Strategic Sales Plans ✕
Marketing Strategy ✕ Key Client Relationships ✕ Customer Loyalty Management ✕
High Performance Sales Teams ✕ Strategic Account Development ✕
Key Account Acquisition & Retention ✕ New Customer Acquisitions ✕ Customer Loyalty ✕
Loyalty Marketing ✕ Direct Marketing ✕ + Add Member Skills

AND also have **ANY** of the following attributes

Interests › Member Interests Search

Member Groups	☐ Arts and Entertainment
Member Interests ›	☐ Business and Management
	☐ Finance and Economy
	☐ Marketing and Advertising
	☐ Politics and Law
	☐ Sales and Retail

Entity Tagging



Member Profile

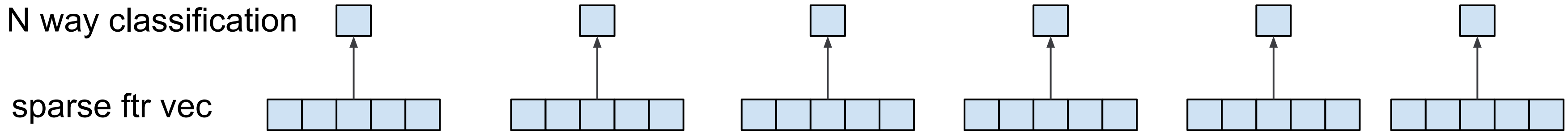
Natural Language Understanding: Entity Tagging

LinkedIn	software	engineer	data	scientist	jobs
CN	T	T	T	T	O
B-CN	B-T	I-T	B-T	I-T	O

B-CN: beginning of a company name
I-CN: Inside of a company name
B-T: beginning of a job title
I-T: Inside of a job title
O: Not an entity
B-PN: beginning of person name
…

Query Tagger: Logistic Regression

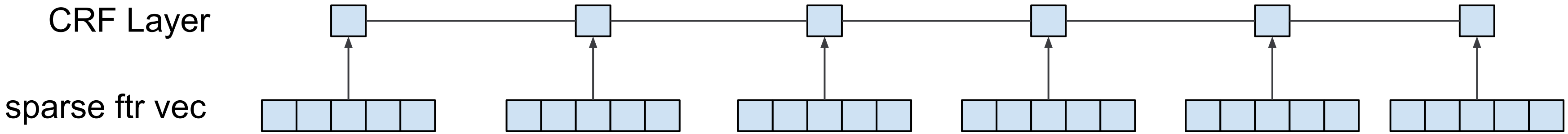
LinkedIn	software	engineer	data	scientist	jobs
B-CN	B-T	I-T	B-T	I-T	O



ftr 0: whether the current word is “linkedin”
ftr 1: whether the current word is “facebook”
…
ftr n: whether the next word is “software”
ftr n+1: whether the next word is “linkedin”
….

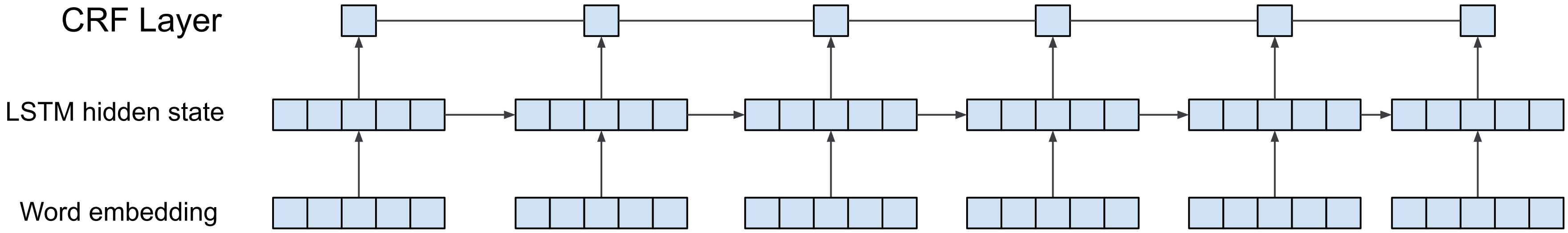
Query Tagger: CRF

LinkedIn	software	engineer	data	scientist	jobs
B-CN	B-T	I-T	B-T	I-T	O

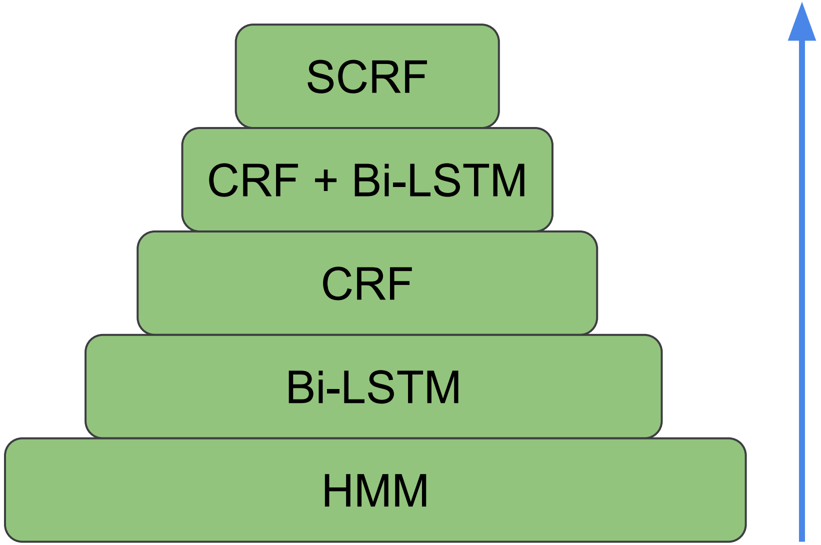


Query Tagger: CRF + LSTM

LinkedIn	software	engineer	data	scientist	jobs
B-CN	B-T	I-T	B-T	I-T	O

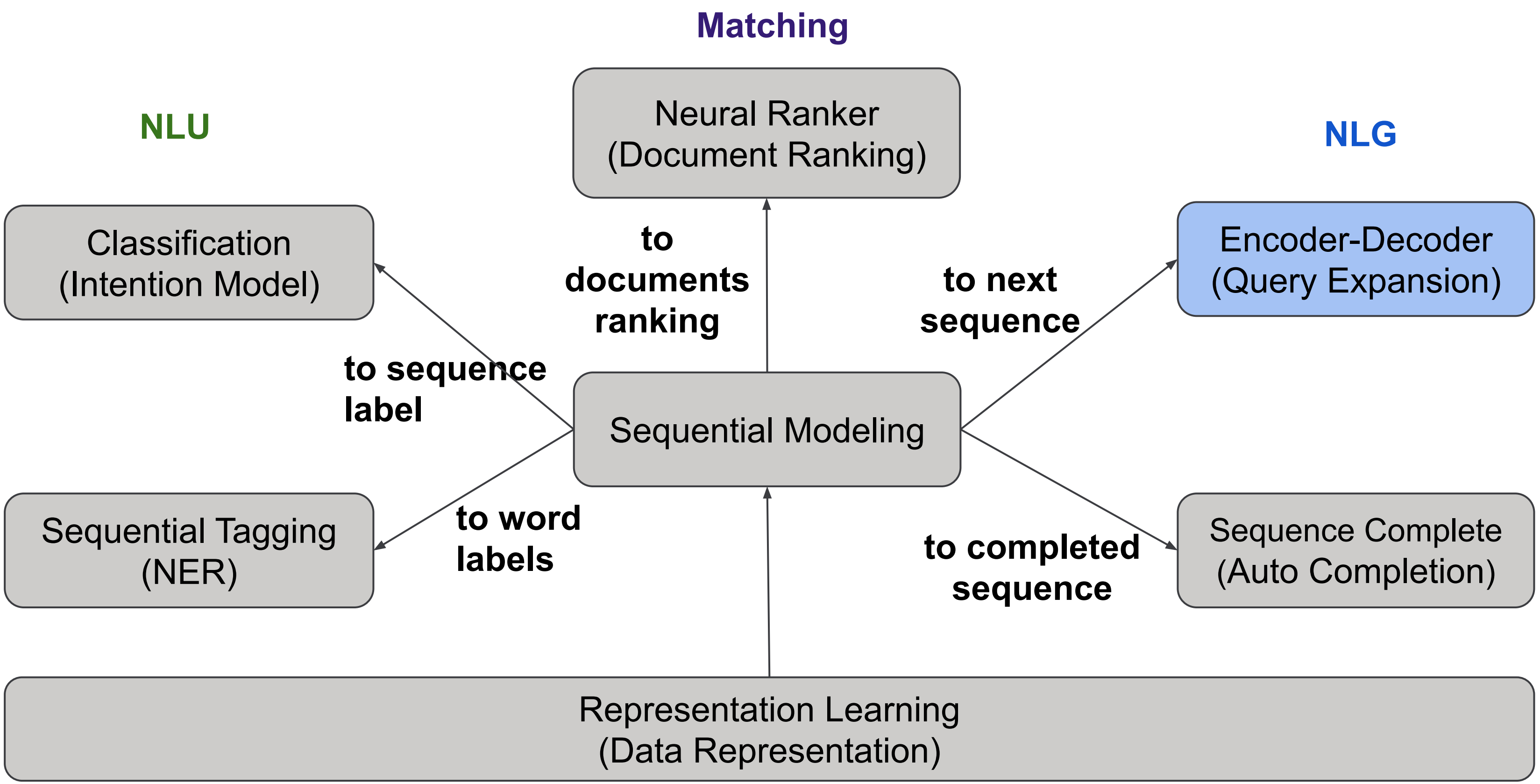


Query Tagger Performance

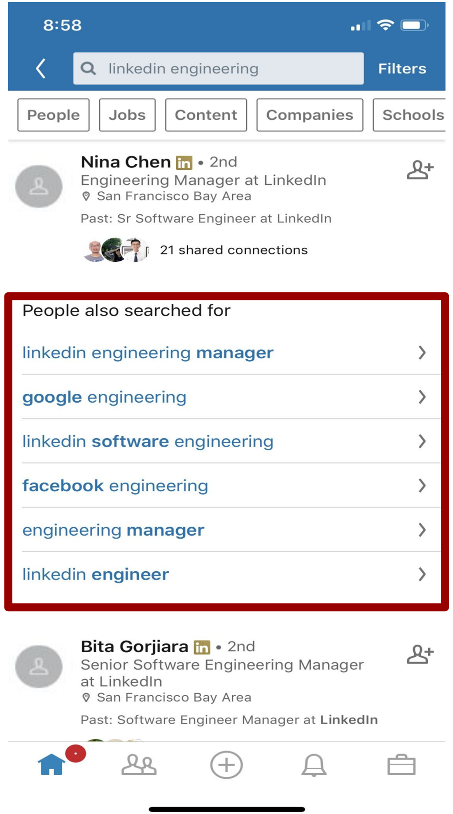


Deep & Wide Modeling Structure for Entity Tagging

Deep Learning for Natural Language Processing

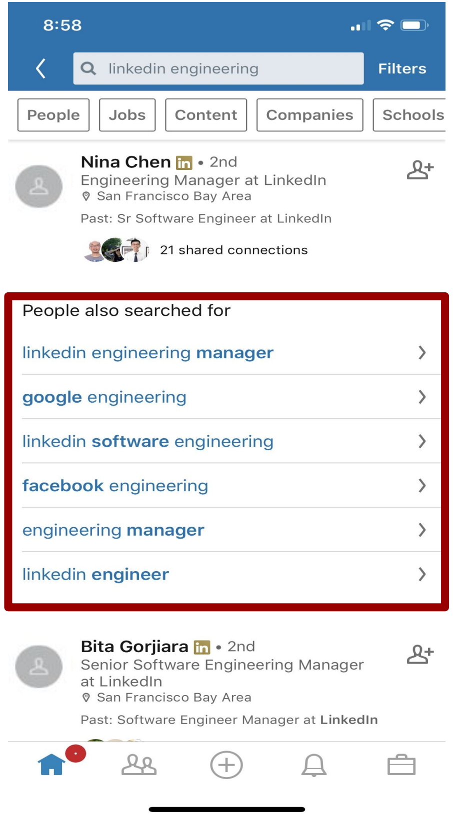


Natural Language Generation: Query Suggestion



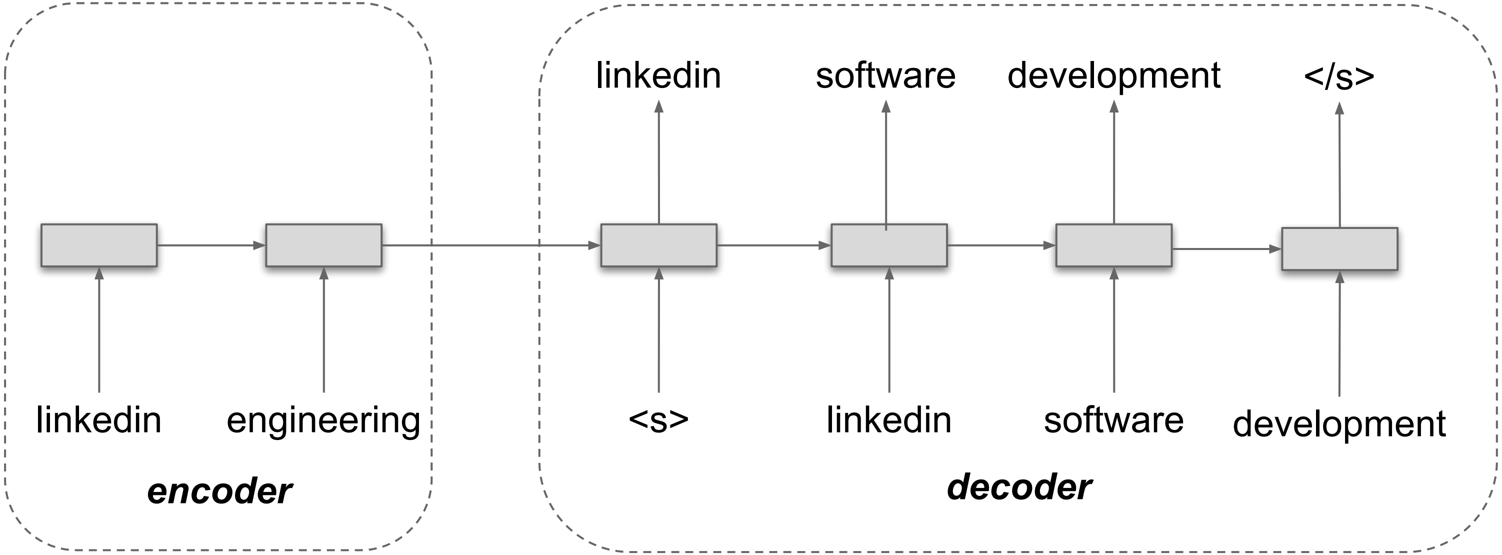
- Save User Effort of Rephrasing Informative Queries
- Capture Users' Search Intention
- Personalization

Natural Language Generation: Query Suggestion



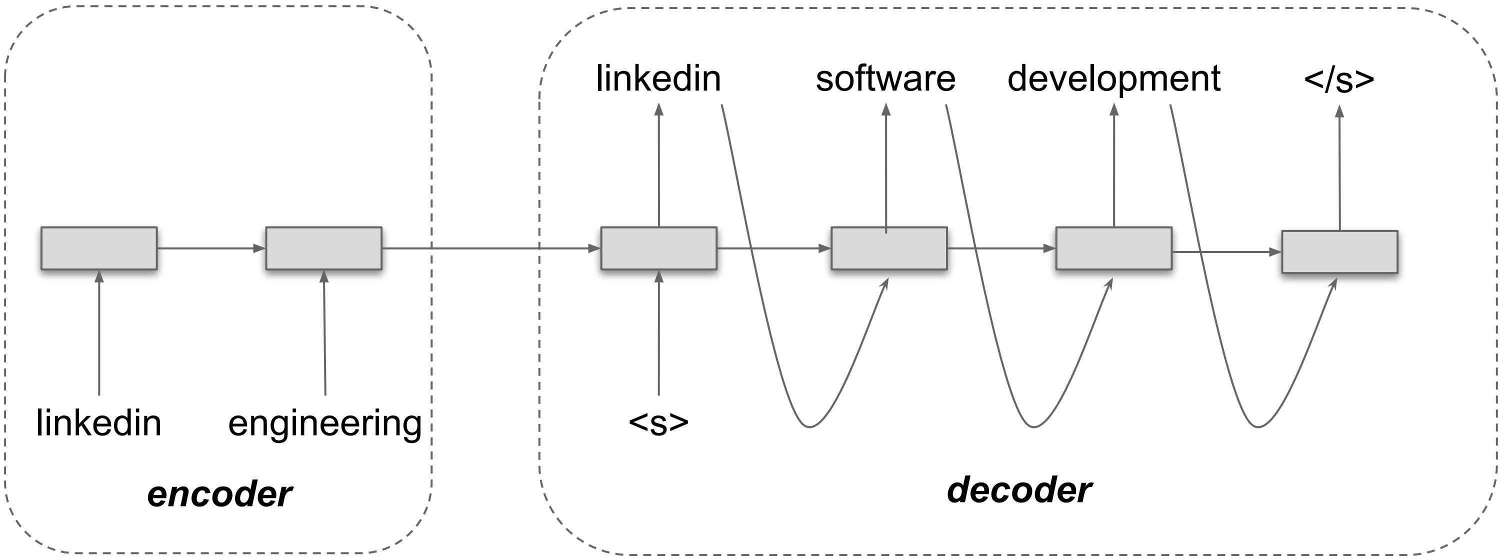
- Traditional Frequency Based Methods
  - Collect <q1, q2> pairs from search log
  - Save the frequent pairs in a key-value store
- Lack of Generalization
  - Purely string matching
  - Cannot handle unseen queries, rare words
- Seq2seq: Model Query Reformulation Behavior

Query Suggestion: Reformulate to Related Queries



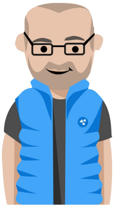
- Training: the 2nd query is given
- Maximize $P(\mathbf{y}|\mathbf{x}) = \prod P(y_i|h_i)$

Query Suggestion: Reformulate to Related Queries

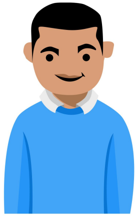


- Inference: the 2nd query is unknown
- Beam search instead of greedy search

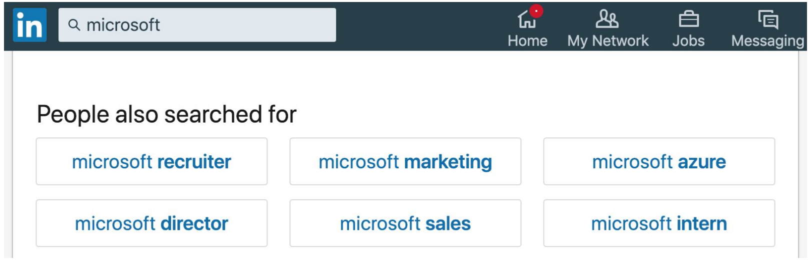
Query Suggestion: Personalization

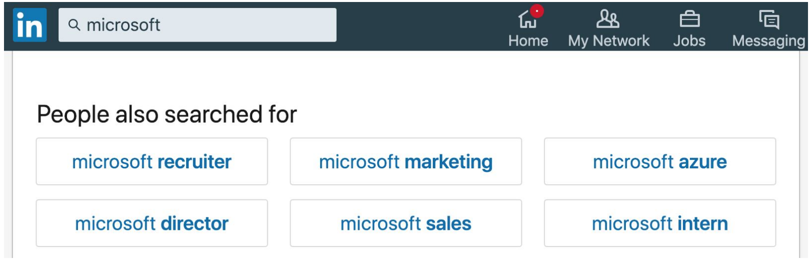


Software engineer

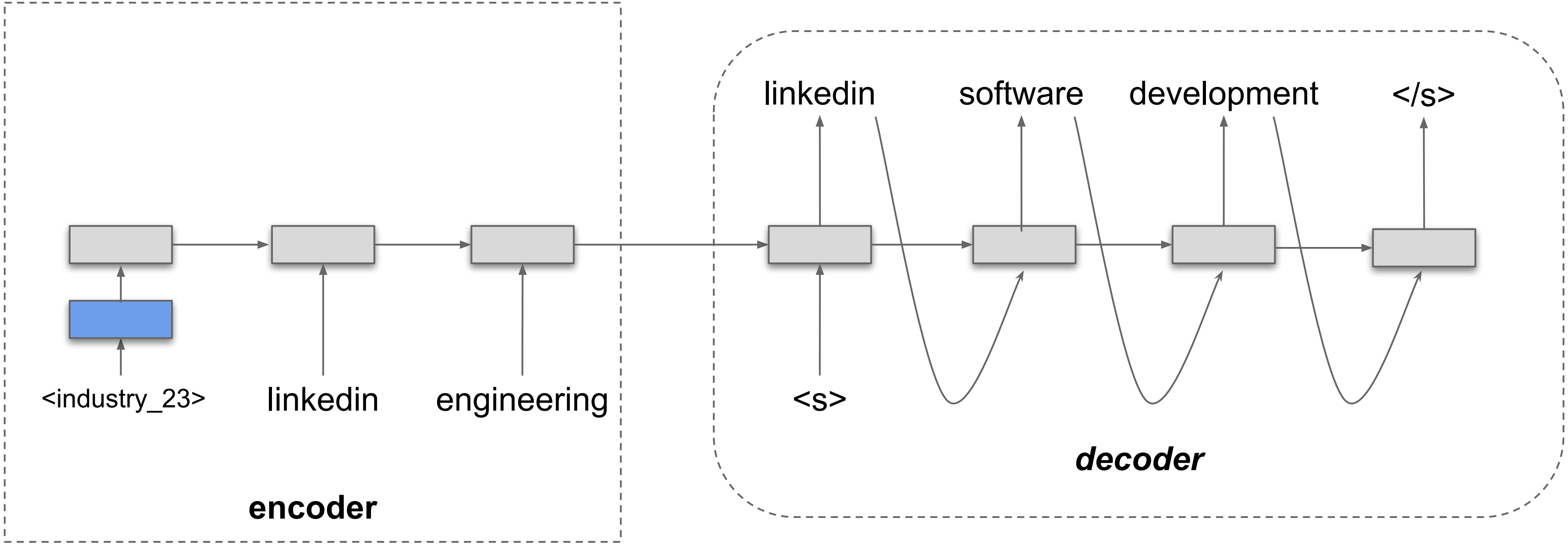


Accountant



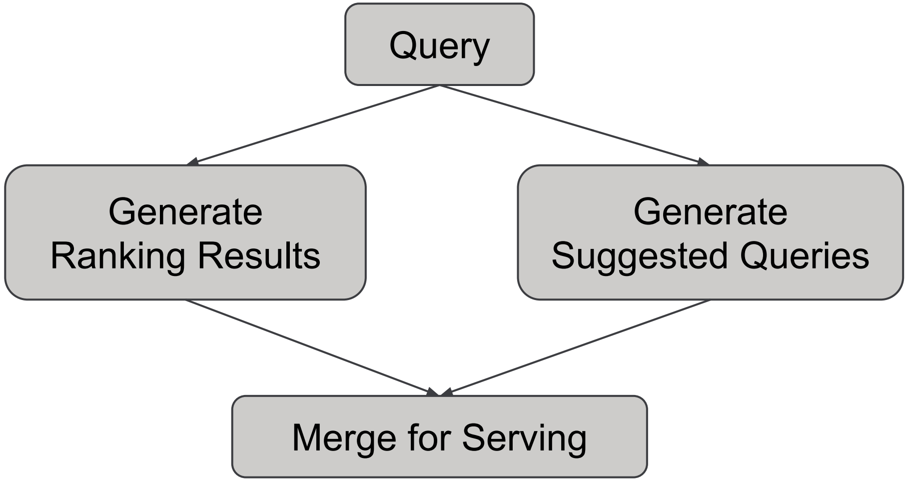


Query Suggestion: Personalization



Query Suggestion: How to Handle Online Latency

- Latency is strictly constrained for one query
  - Make it parallel with search ranking



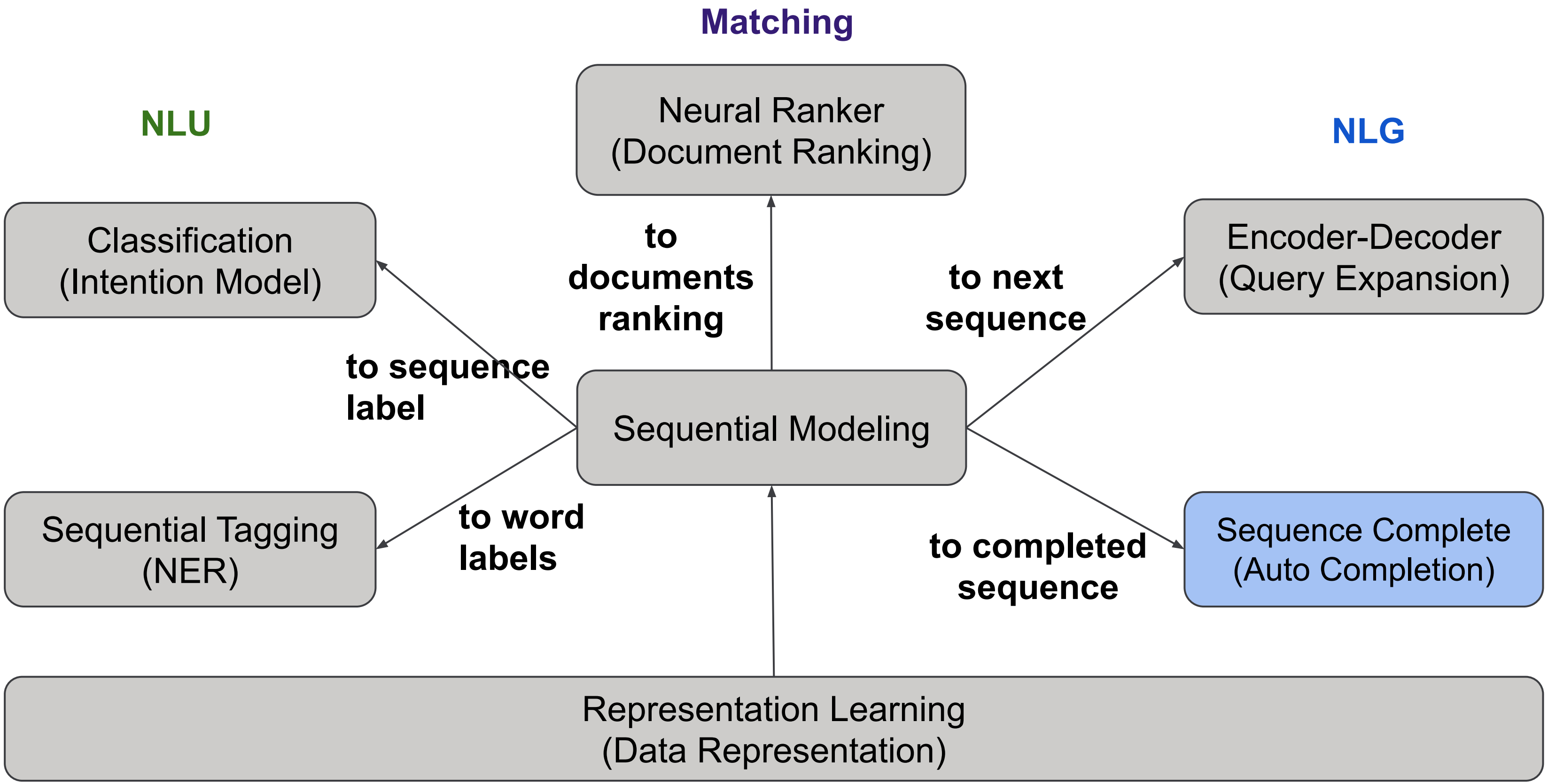
Online Performance

English Market

- No Personalization
  - Coverage: +80% Impressions, +26% CTR
  - +1% Total job application

- With Personalization
  - +1.18% Session Success for passive job seekers

Deep Learning for Natural Language Processing

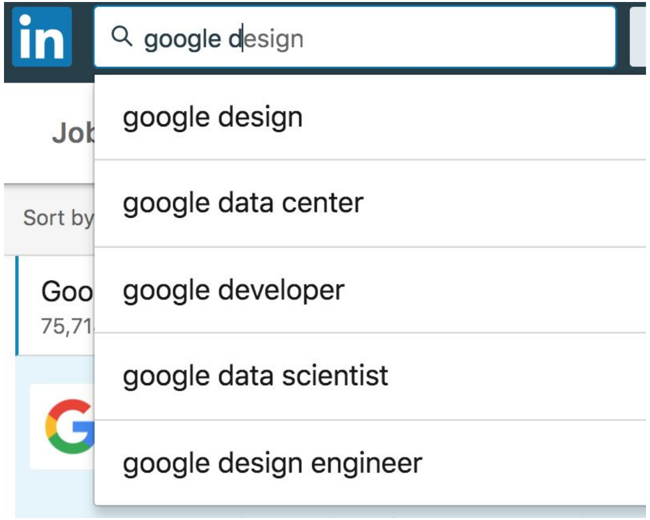


Natural Language Generation: Auto-Completion

- Given a prefix, predict the **completed query**, rather than the **completed word**

Auto-completion Challenges

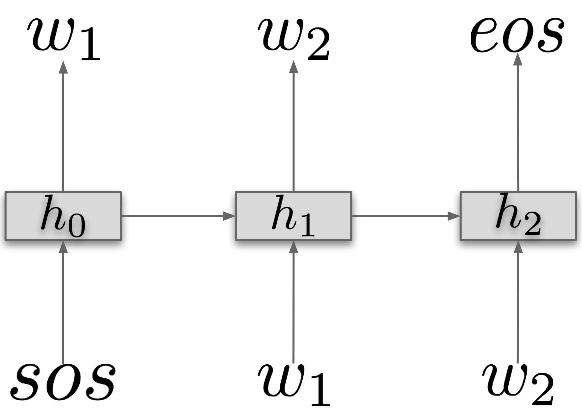
- **Short Context**
  - How to enrich semantic features
- **Personalization**
- **Latency Restriction**
  - Have to adopt simple and fast models



A Two-step Approach: Generation and Ranking

- **Candidate Generation**
  - Collect query frequency from search log
- **Candidate Ranking**
  - Neural Language Model serves as a scoring function

Auto-Completion: Neural Language Model as Scoring/Ranking



$$s(q) = \sum_i \log P(w_{i+1}|h_i)$$

- Achieved 6x speedup with optimization

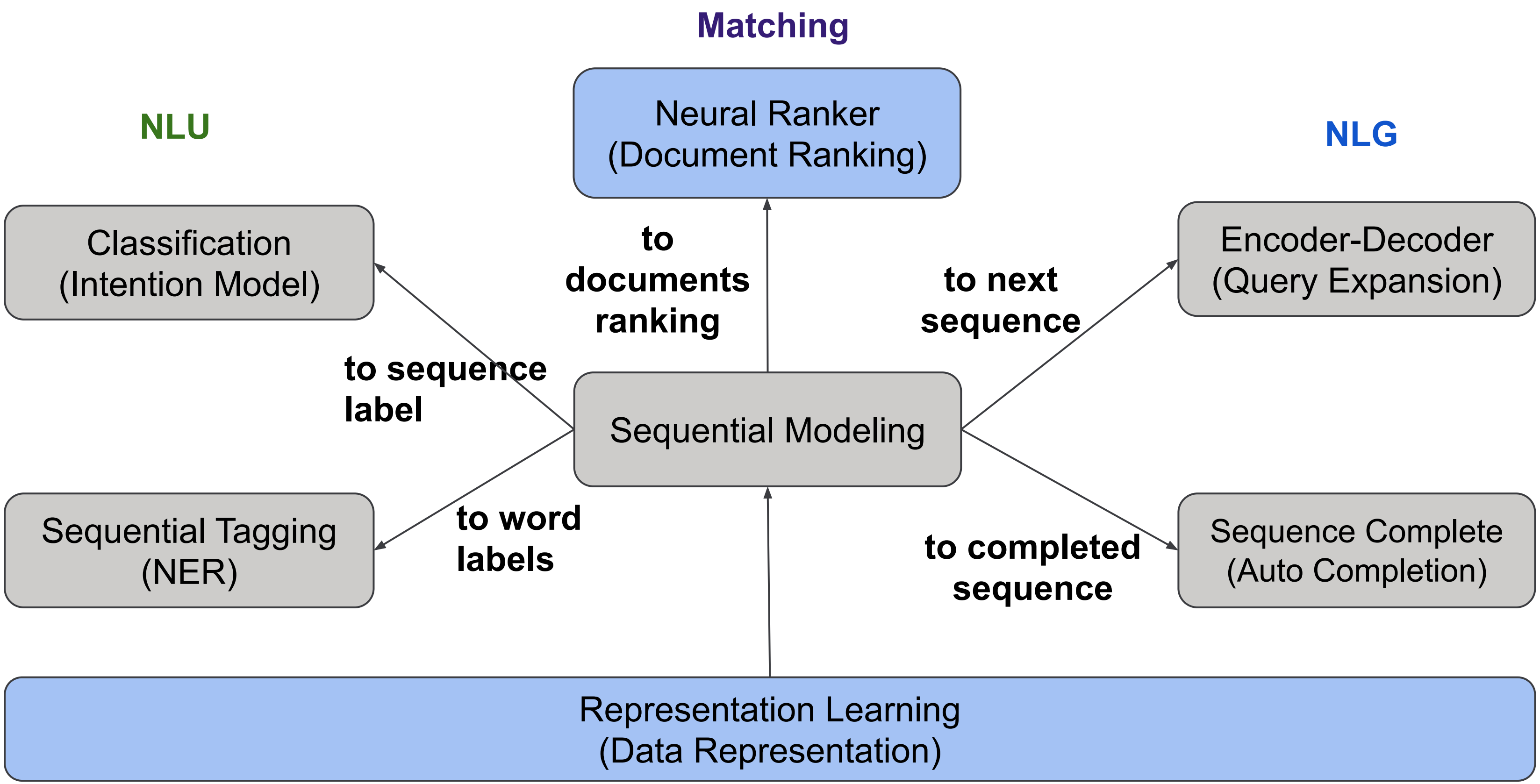
Auto-Completion Online Experiments

Neural Auto-completion vs Frequency Based Methods

- People Search English Market
  - +88.75% autocomplete clicks
  - +0.81% CTR@1 in SERP

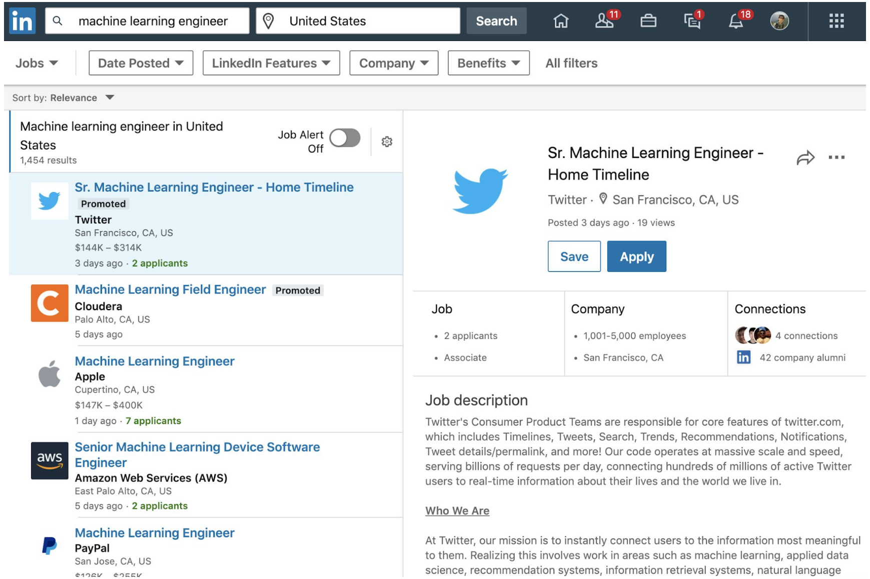
- Job Search German Market
  - +3.24% Results clicked
  - -6.24% Raw searches w/o results
  - +3.24% Entity view

Deep Learning for Natural Language Processing

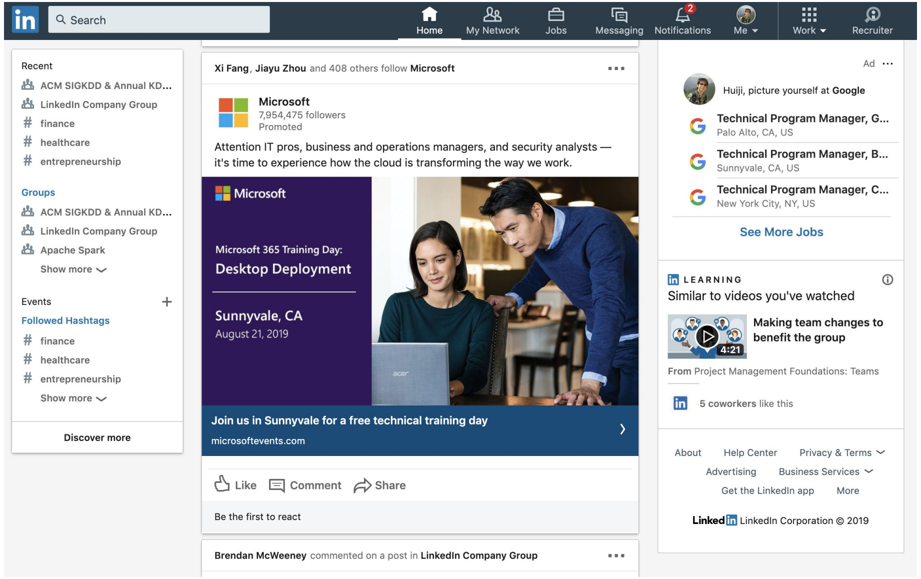


Natural Language Processing: Document Ranking

Job Search



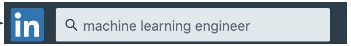
Ads Recommendation



Natural Language Processing: Document Ranking

Search Systems

Query



Rank

$P\,(\text{match} \mid \text{user, query, doc1})$

$P\,(\text{match} \mid \text{user, query, doc2})$

$P\,(\text{match} \mid \text{user, query, doc3})$

……

$P\,(\text{match} \mid \text{user, query, docN})$

Recommender Systems

Rank

$P\,(\text{match} \mid \text{user, doc1})$

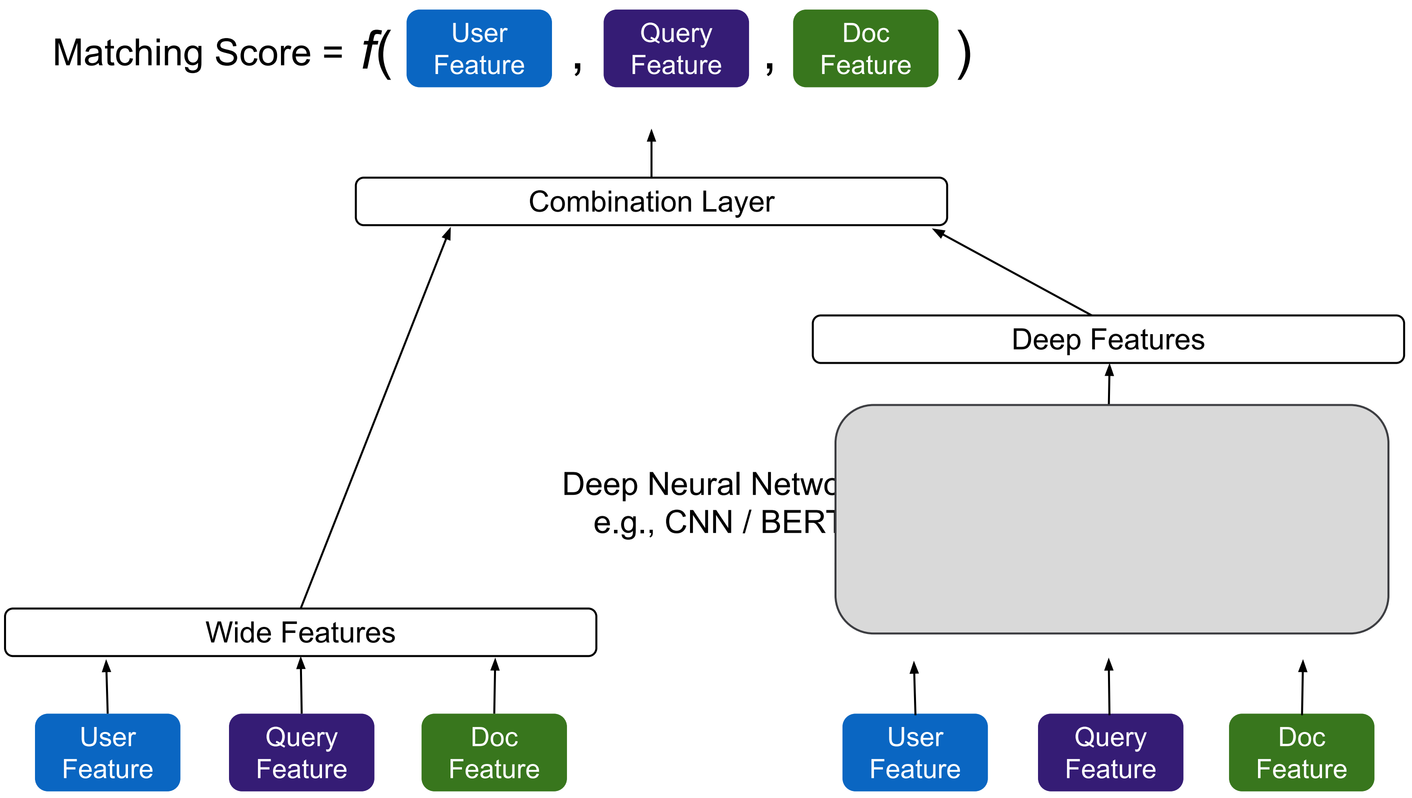
$P\,(\text{match} \mid \text{user, doc2})$

$P\,(\text{match} \mid \text{user, doc3})$

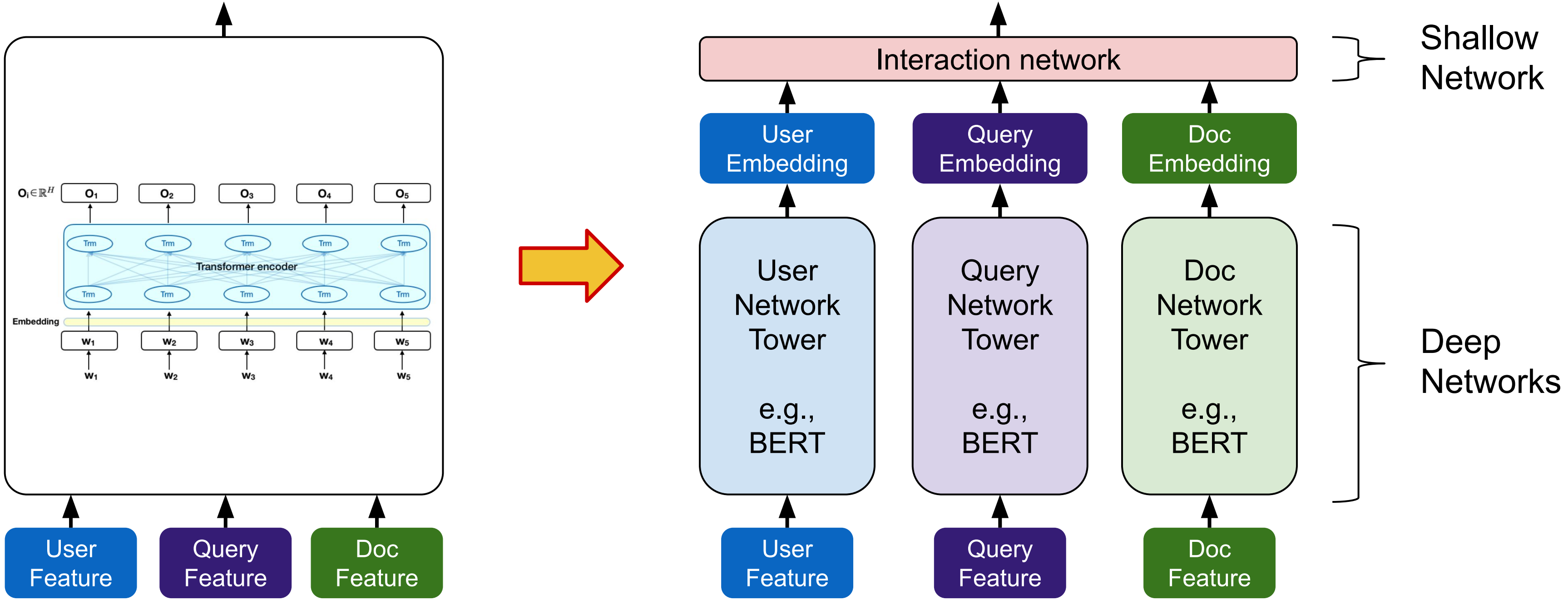
……

$P\,(\text{match} \mid \text{user, docN})$

Natural Language Processing: Document Ranking



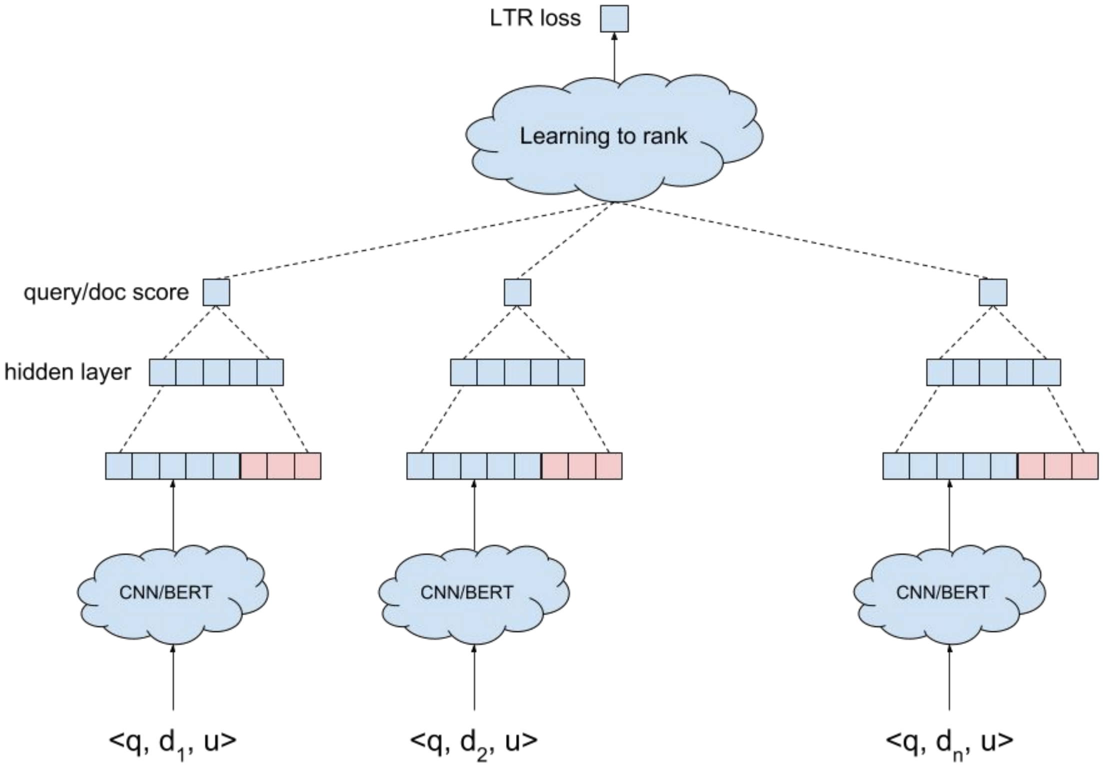
Natural Language Processing: Document Ranking



One Big Network

Multi-Tower Network

Neural Ranking



Experiments

- Offline

People Search (NDCG@10)		Job Search (NDCG@10)	Ads Click-through Rate (AUC)	Ads Conversion (AUC)
Wide Features	-	-	-	-
CNN	+1.32%	+2.36%	+3.43%	+1.37%
LinkedIn BERT	+1.98%	+5.14%	-	-

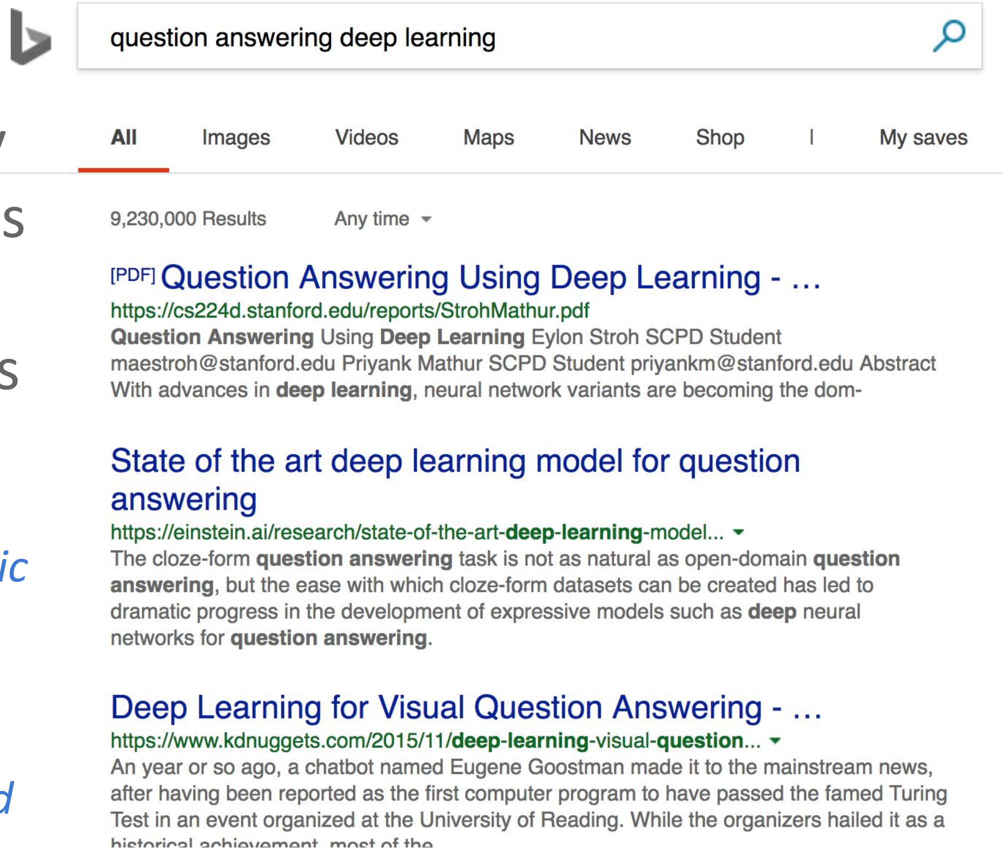
- Online (CNN)
  - **Job Search**
    - Job apply clicks: **+1.62%**
    - Onsite paid applicants: **+1.38%**
  - **People Search**
    - Satisfied clicks: **+0.78%**
    - Successful sessions: **+0.37%**

Deep NLP in Search and Recommender Systems - Question Answering at LinkedIn

Liang Zhang

Traditional Web Search

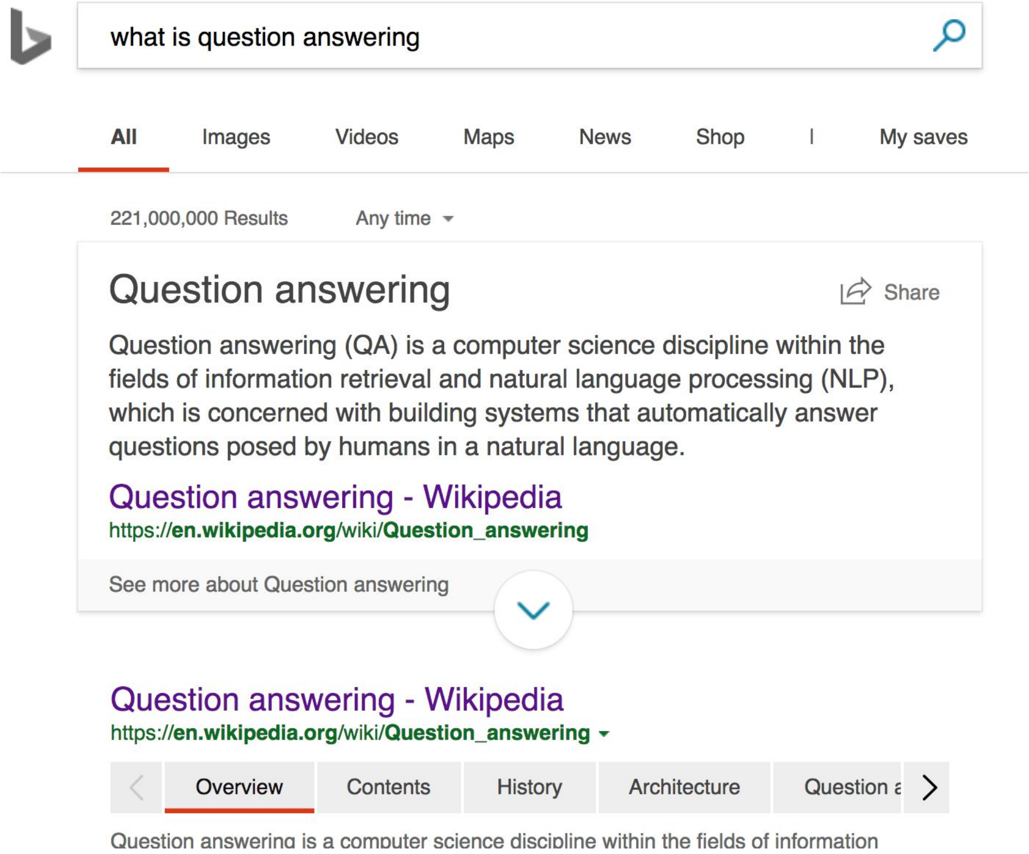
- User Input: Keyword query
- Output: A list of documents
- A classic way of obtaining information from machines



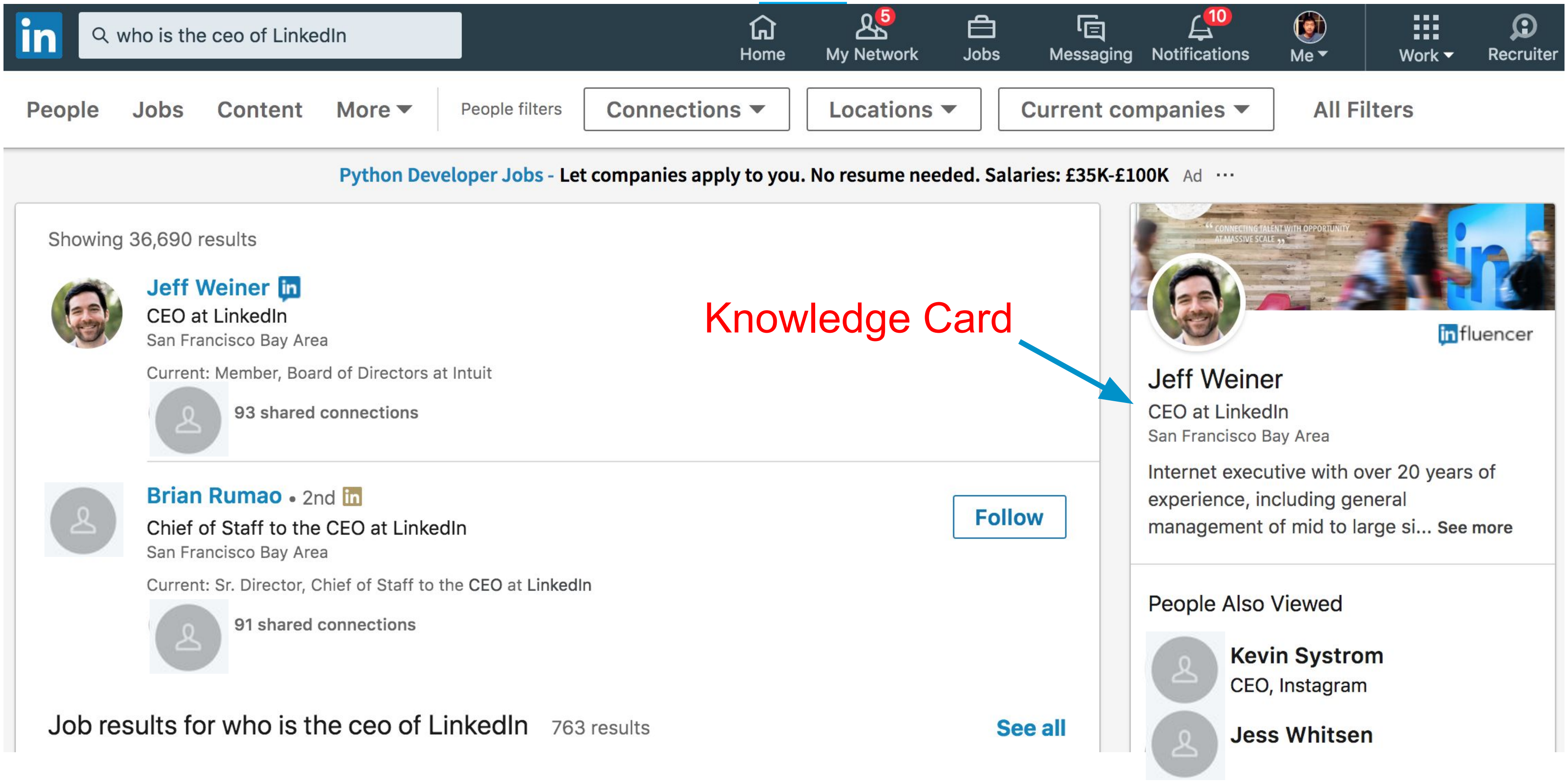
What can be done better:

- *Understand user intent and semantic meaning of the query*
- *Directly show the answer*
- *Interact with the users when the results do not satisfy the user's need*

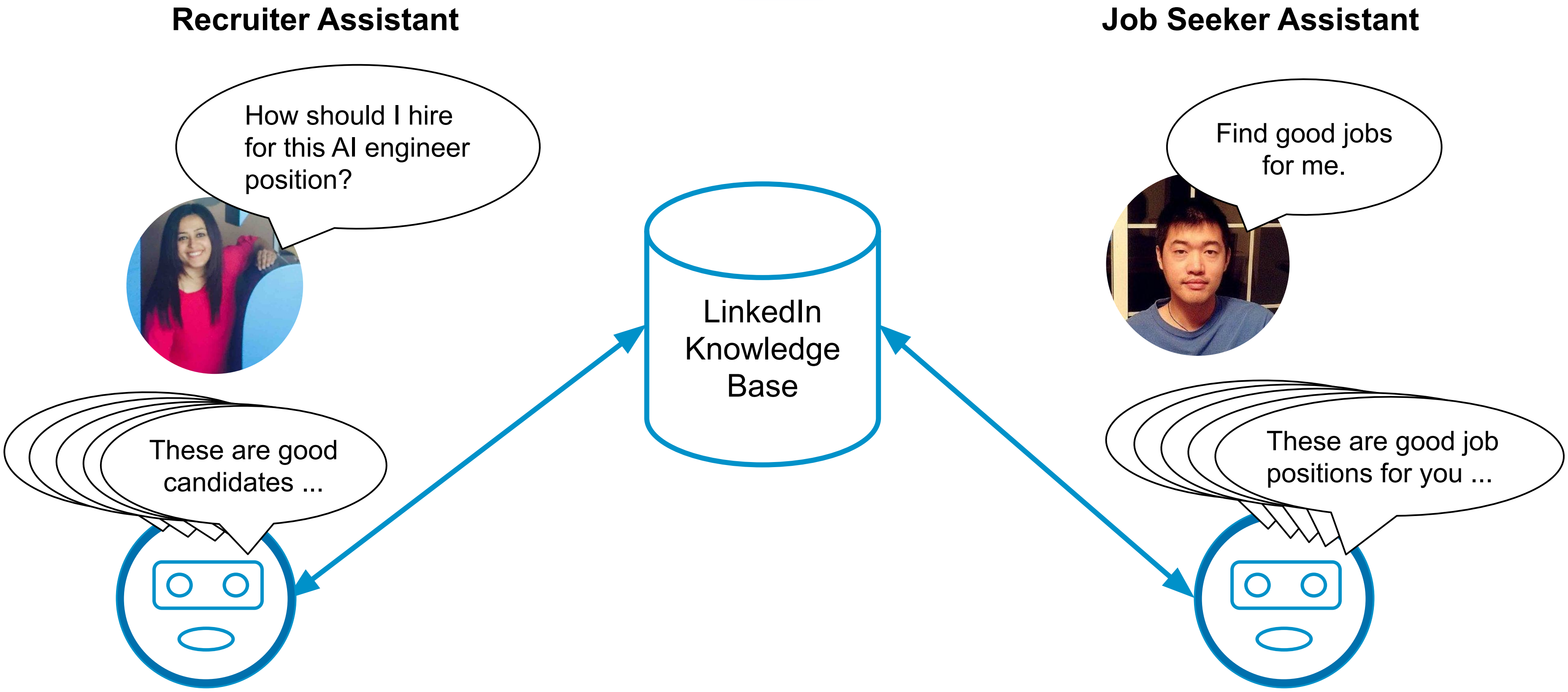
Modern Web Search with Question Answering



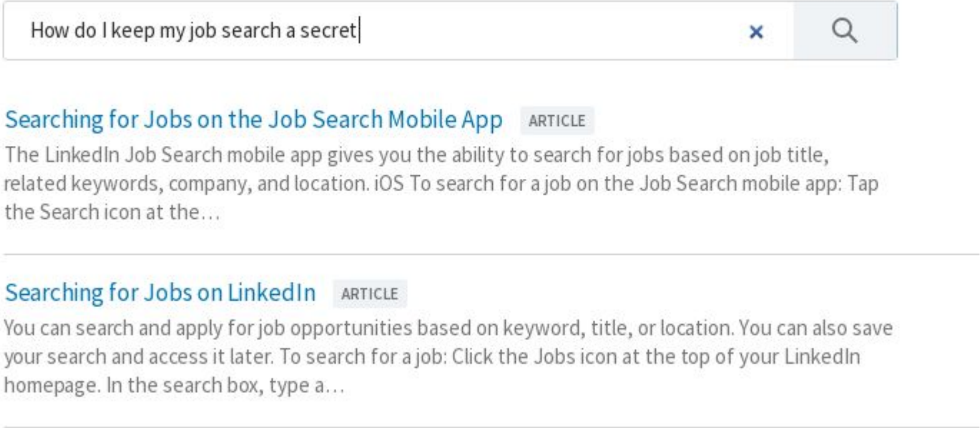
Modern Web Search with Question Answering



How Question Answering Can Help LinkedIn



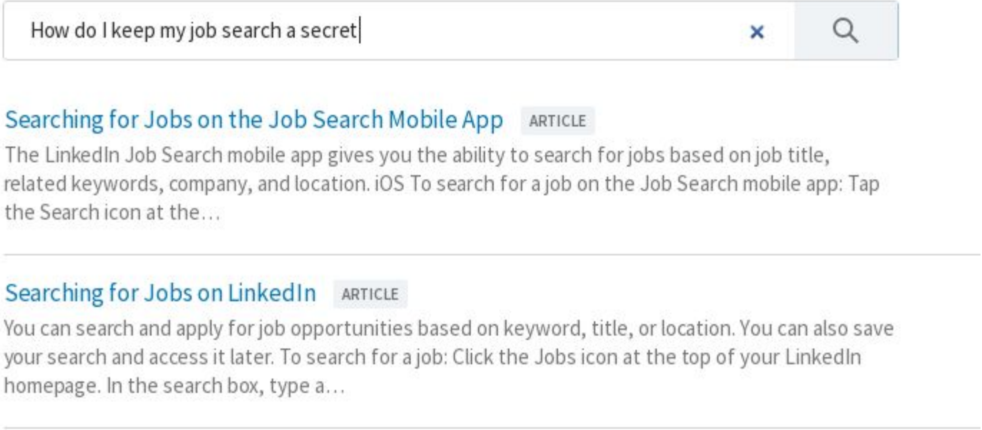
LinkedIn Help Center



- Input: Question in Natural Language
- Traditional keyword search does not work well
- Our solution:
    - Question-answering through Search
    - Chatbot (Demo)

Question Answering Through Search

LinkedIn Help Center Search

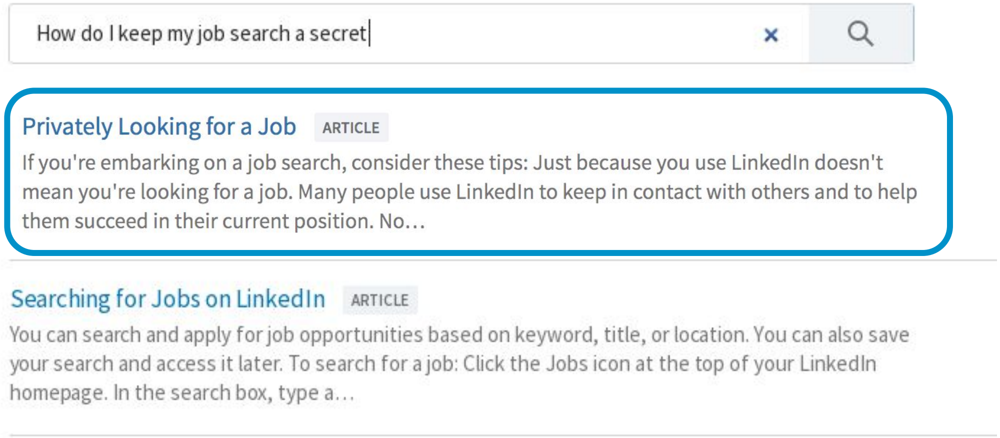


- Input: Question as a Query
- Output: Article / Section of the Article that contains the answer
    - Few thousands candidates
    - New candidates get added regularly

Our Data

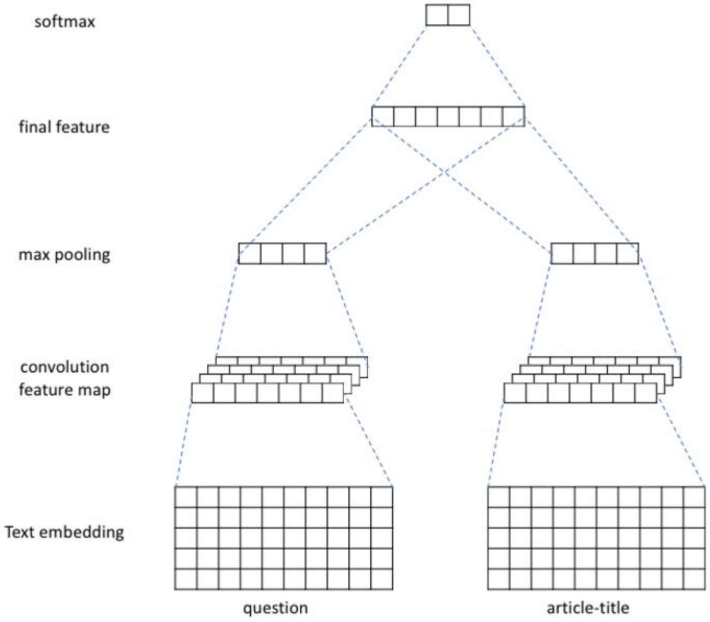
- A random sample of user click-through data in 01/2017 - 04/2018
- 200K <query, list of articles> samples, click / non-click as responses for each <query, article>
- Only contains queries with > 4 words (*where traditional keyword search generally fails!*)

Our Modeling Approach



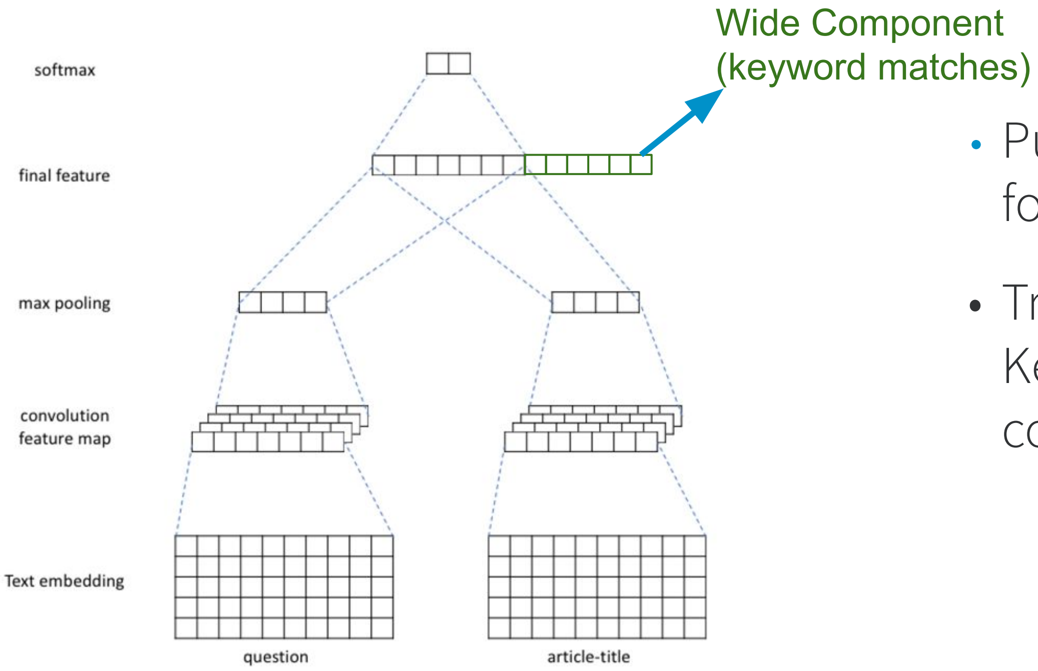
- Consider each candidate article as an intent
- Problem: Natural language Query => Intent
- Need to handle regularly added new articles without too-frequently retraining the model

Our Modeling Approach -- CNN



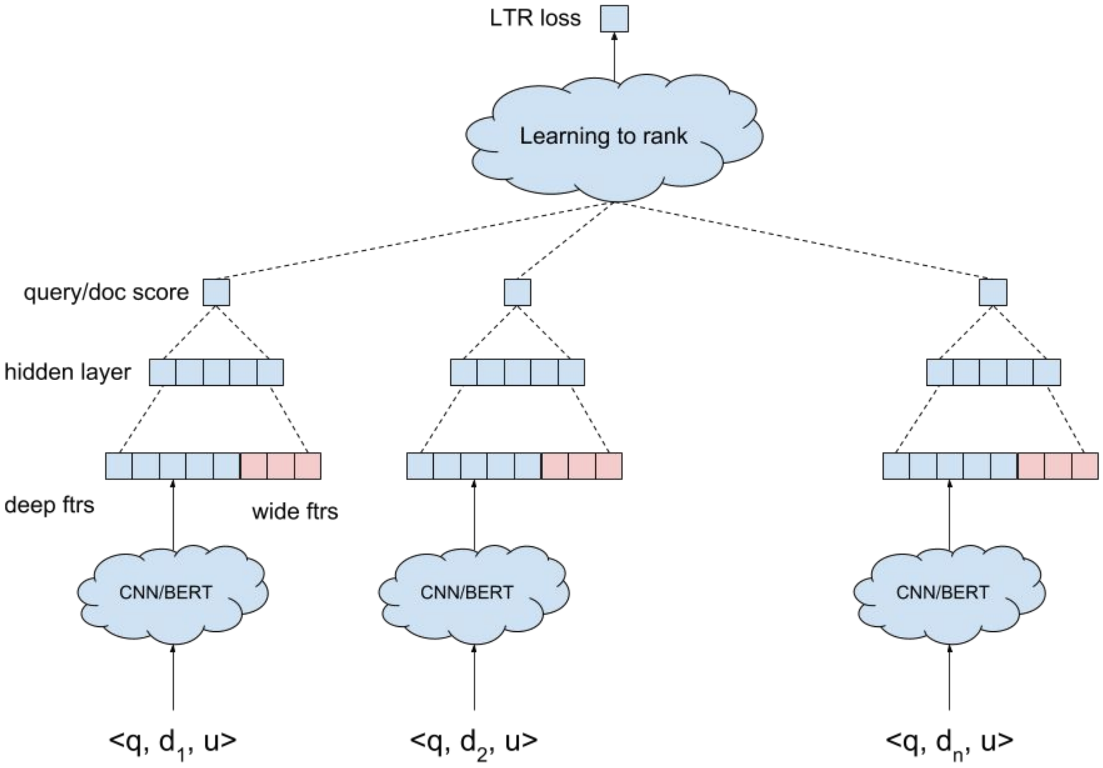
- Independent CNN layers for Question & article title, keywords, content
    - Word2Vec Embedding to initialize
- Final layers: Sim<Question Embedding, Article Embedding> => Softmax over all candidate questions
- For new articles in test data: Use embedding similarity instead of retraining the model

Deep + Wide Model with Better Performance



- Pure CNN Model did not work well for out-of-vocabulary queries
- Traditional IR-based Keyword-match features added to complement (wide component)

Our Modeling Approach -- BERT + Wide



Offline Evaluation

Model	Lift % in Precision@1 vs Control (IR-based)
Deep Model (CNN)	+26%
CNN + Wide Model	+32%
BERT + Wide Model	+36%

Online A/B Test Performance

Metric	CNN vs Control	BERT vs CNN
Happy Path Rate	+15%	+11.3%
Clicked Session	+5.7%	+5.2%

- Happy Path: Users who clicked only one search result and then left help center without creating a case

More Examples...

LinkedIn Help

how to download my resume

Saving a Profile in a PDF Format ARTICLE

You can save a copy of your profile or someone else's profile in a PDF format from the introduction card on your profile. Note: We currently don't offer the option to choose which sections will be included in saved versions of your profile. To…

Uploading Your Resume when Applying for Job on LinkedIn ARTICLE

If you're applying for a job through LinkedIn and not through a company's web site, you can attach your resume to the application in addition to applying with your LinkedIn profile. To upload a new resume: Search for a job. Click on a job…

Using LinkedIn Resume Builder (currently available in India only) ARTICLE

Important: Resume Builder is currently available to only LinkedIn members in India who've subscribed to Premium Career. Please note that this new feature will be gradually rolled out to our members and may not be available to you at this time…

Troubleshooting Learning Video Downloads and Mobile Viewing ARTICLE

If the video you're streaming is pausing, or you're unable to download a video on your mobile device, you may be experiencing a problem with your network connection. While you can stream or

LinkedIn Help

I am a recruiter, how to find good candidates?

Finding Additional Job Candidates ARTICLE

You can find additional job candidates using various features of LinkedIn. Starting from basic search to Recommended Matches and LinkedIn Recruiter Lite, LinkedIn offers many ways to help you find the best candidates for your open job…

Differences Between Recruiter, Recruiter Premium Subscription (RPS), and Recruiter Lite ARTICLE

While Recruiter Lite lets you search for, filter, and contact LinkedIn members who may be good candidates for your role and comes with InMail, a Recruiter account gives you unrivaled access to the entire LinkedIn network. Designed specifically for…

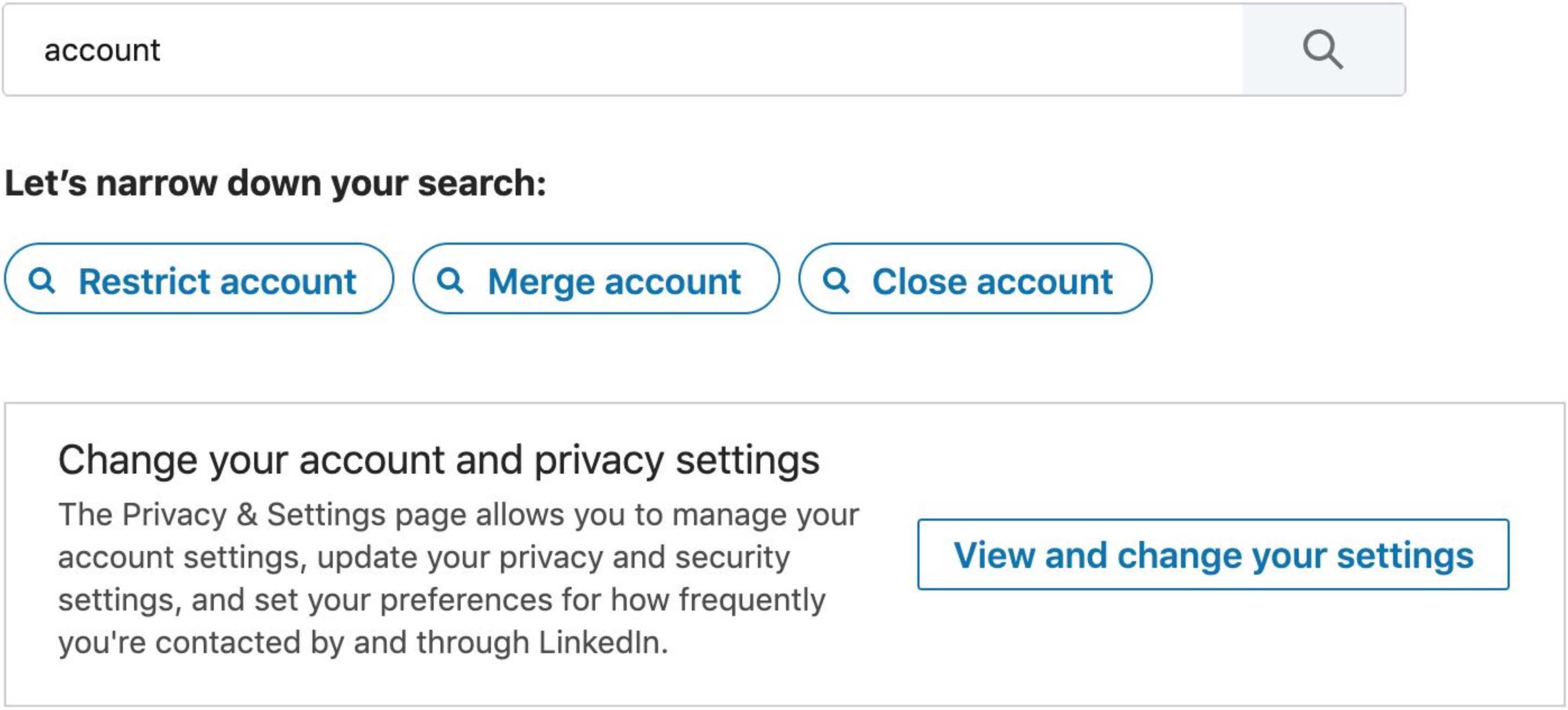
Best Practices for Posting Jobs ARTICLE

Recruiters can find the best candidates for the open job role faster and easier with effective use of all the LinkedIn features. We recommend the below best practices to enhance your job posting experience: Creating and maintaining the…

Targeting Options and Best Practices for LinkedIn Advertisements ARTICLE

When advertising on LinkedIn, you can get your message in front of the right people when they are most engaged by using targeting. You can deliver your content to those who matter most to your

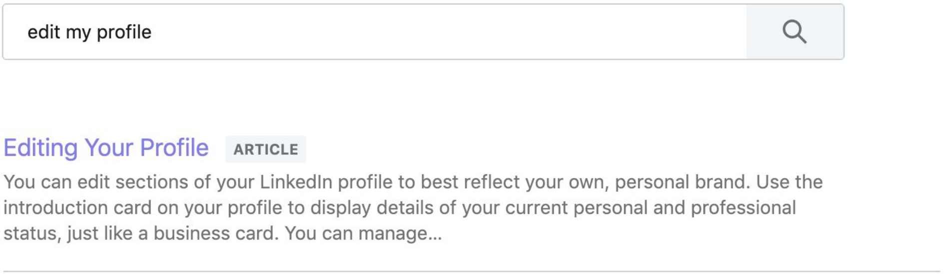
Interactive Search (Demo)

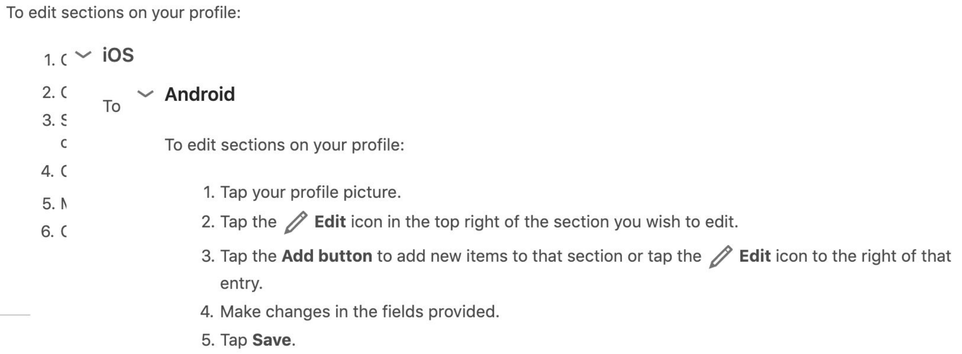


ChatBot for LinkedIn Help Center

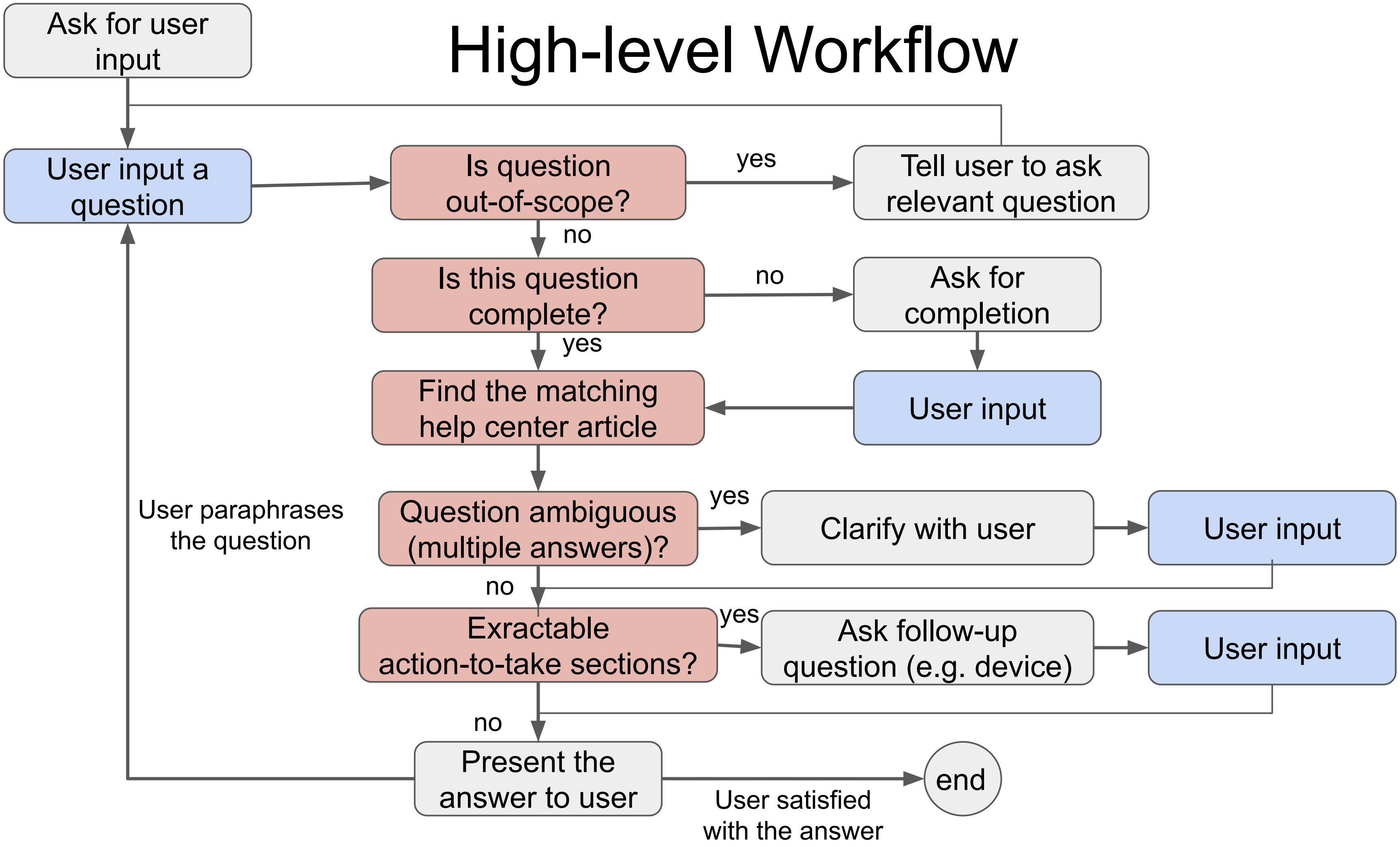
Problems to Solve

- Out-of-scope questions
  - “How are you?”
  - “Will you marry me?”
- Ambiguity
  - Incomplete questions (e.g. verb / object missing)
    - e.g. “premium” => “cancel premium”? “be premium”?
  - Question is complete (verb + object) but ambiguous
    - e.g. “Cancel account” => “cancel premium subscription”? “closing LinkedIn account?”
- Actions to take instead of just article link





High-level Workflow



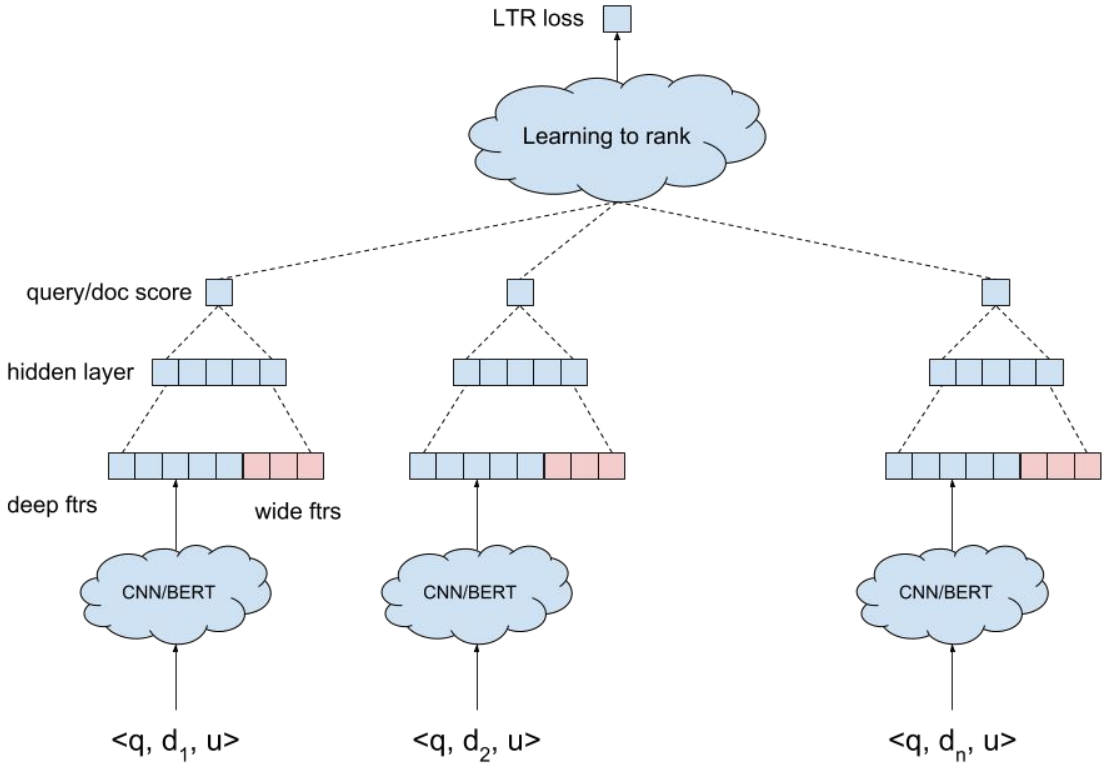
Out-of-Scope Detection Model

- *“Who is abraham lincoln?”* → Not help center query
- Current: A binary classification approach
  - Positive: help center queries
  - Negative: LinkedIn search queries, movie dialogue, quora questions
- Future:
  - Being able to respond to other intents, e.g.
    - Greetings
    - Ask for speaking to an agent
    - Looking up personal account info (has my premium account expired?)

Incomplete Question Handling

- Incomplete Questions:
    - Object (noun) missing
        - e.g. “cancel” -> what to cancel?
    - Verb missing
        - e.g. “Profile” → what to do with profile?
- We assume the question must contain nouns (objects) and verbs
    - If not, explicitly ask for it
- Standard POS Tagging model from openNLP
- If incomplete: Suggest most related & frequent complete queries from query logs
- Future work
    - Use Entity Tagging in help center context to detect question completeness

Question-Answering Model (BERT + Wide)



Actions-to-take Section Extraction & Ambiguity Handling

- Ambiguity Handling
  - Clarify with user if scores among top 3 articles are too close
- Extract Action-to-take Sections from Top-ranked Article
  - Pattern mining (e.g. ios / android section heads) + Human labeling
  - Follow up with user to pick the right section with actions to take (e.g. desktop / ios / android)

Demo

Conclusion

Lessons & Future Trends

- **Ground Truth Data**
  - Human Labelled Data (Crowdsourcing)
  - User Behavior Data
  - Automatic Data Generation with Generative Models (e.g., GANs)

- **Efficient Modeling Training and Serving**
  - Pre-trained Model for Multiple Products
  - Automatic Hyperparameter Tuning & Structure Learning
  - Memory and Latency Optimization for Efficient Online Serving

Lessons & Future Trends (Cont'd)

- **Internationalization**
  - Transfer Learning
  - Multilingual Learning

- **Reinforcement Learning**
  - Long-term reward optimization in near-real-time user feedback loop
  - Multi-turn question answering

- **Model Debugging**
  - Model Interpretation
  - Model Fairness & Privacy

Thank You!