TEXT MINING

L06. INFORMATION EXTRACTION

SUZAN VERBERNE 2021

Universiteit Leiden

ASSIGNMENT 1 – TEXT CLASSIFICATION

- If you didn’t make the deadline this week, you can take the resit

	Deadline	Re-sit deadline
Assignment 1	18 October	16 January (maximum grade 6)
Assignment 2	15 November	16 January (maximum grade 6)
Final assignment	16 January	6 February (maximum grade 6)
Written exam	13 January	4 February

- Weight of assignment 1: 10% of total course grade

TODAY'S LECTURE

- Quiz about week 5
- Named Entity Recognition
  - Feature-based models
  - Neural models
- Relation extraction

QUIZ ABOUT WEEK 5

- Why should we have multiple human annotators if we create labelled data?
  a. Because we need to estimate the reliability of the data
  b. Because we need to measure the inter-rater agreement between the annotators
  c. Because there is human interpretation involved in the annotation
  d. All of the above

QUIZ ABOUT WEEK 5

- What is the interpretation of Kappa=0?
  a. No agreement
  b. Complete agreement
  c. Measured agreement equal to expected agreement
  d. Undefined

GENERAL QUESTION

- Are you taking the course ‘Introduction to Deep Learning’?
  a. Yes (or took it last year)
  b. No, but I took another course on neural networks and deep learning
  c. No

INFORMATION EXTRACTION

INFORMATION EXTRACTION

- "Information extraction from text is an important task in text mining. The general goal of information extraction is to discover structured information from unstructured or semi-structured text."
- Example applications:
  - automatically identify mentions of biomedical entities from patents and to link them to their corresponding entries in existing knowledge bases
  - find person names in bank transactions/electronic health records for the purpose of anonymization
  - find company names, dates and stock market information in economic newspaper texts
  - More advanced search problems such as entity search, structured search and question answering can provide users with better search experience

INFORMATION EXTRACTION TASKS

- Named Entity Recognition (NER)
(sections 8.3, 8.4, 8.5 in J&M)
- Relation extraction
(sections 17.1 and 17.2 in J&M)

NAMED ENTITY RECOGNITION

RECOGNIZING ENTITIES

- A named entity is a sequence of words that designates some real-world entity (typically a name), e.g. 'California', 'Steve Jobs' and 'Apple Inc.'
  - General types, occurring in most domains: person, organization, location
  - Extended types (no names): dates, times, monetary values and percentages
  - Domain-specific types, e.g. biomedical entities
- Task: Named entity recognition (NER)

CHALLENGES OF NER

- Ambiguity of segmentation:
  - where are the boundaries of an entity? (e.g. ‘King Willem-Alexander of the Netherlands’)
- Type ambiguity
  - E.g. The mention ‘JFK’ can refer to a person, the airport in New York, or any number of schools, bridges, etc.
- Shift of meaning
  - E.g. ‘president of the US’ refers to Donald Trump, but in a newspaper article from 2011 it refers to Obama

RECOGNIZING ENTITIES

- We could use a list of names and label them in the text. Limitations?

RECOGNIZING ENTITIES

- We could use a list of names and label them in the text. Limitations:
  - Entities are typically multi-word phrases (boundaries?)
  - List is limited (new names, new domains)
  - We would need to add all variants (Trump, Donald Trump, Donald John Trump, President Trump, Mr. Trump, …)

RECOGNIZING ENTITIES

- We could use a list of names and label them in the text. Limitations:
  - Entities are typically multi-word phrases (boundaries?)
  - List is limited (new names, new domains)
  - We would need to add all variants (Trump, Donald Trump, Donald John Trump, President Trump, Mr. Trump, …)
- NER is a sequence labelling problem
  - a word-by-word sequence labelling task, in which the assigned tags capture both the boundary and the type

SEQUENCE LABELLING

- NER is a sequence labelling task
  - sequence = sentence; element = word; label = entity type
  - one label per word
- Format of training data: IOB tagging
  - Each word gets a label (tag)
  - beginning (B), inside (I) of each entity type
  - and one for tokens outside (O) any entity

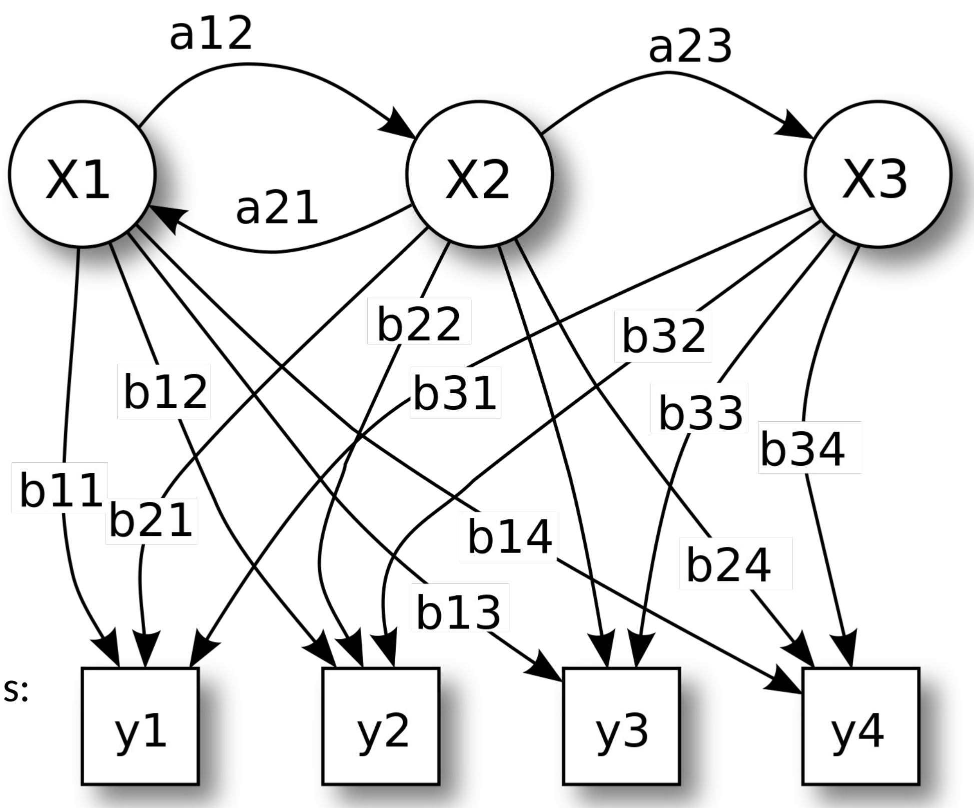
Words	IOB Label
American	B-ORG
Airlines	I-ORG
,	O
a	O
unit	O
of	O
AMR	B-ORG
Corp.	I-ORG
,	O
immediately	O
matched	O
the	O
move	O
,	O
spokesman	O
Tim	B-PER
Wagner	I-PER
said	O
.	O

SEQUENCE LABELLING MODELS

J&M CHAPTER 8

HIDDEN MARKOV MODEL (HMM)

States:
labels



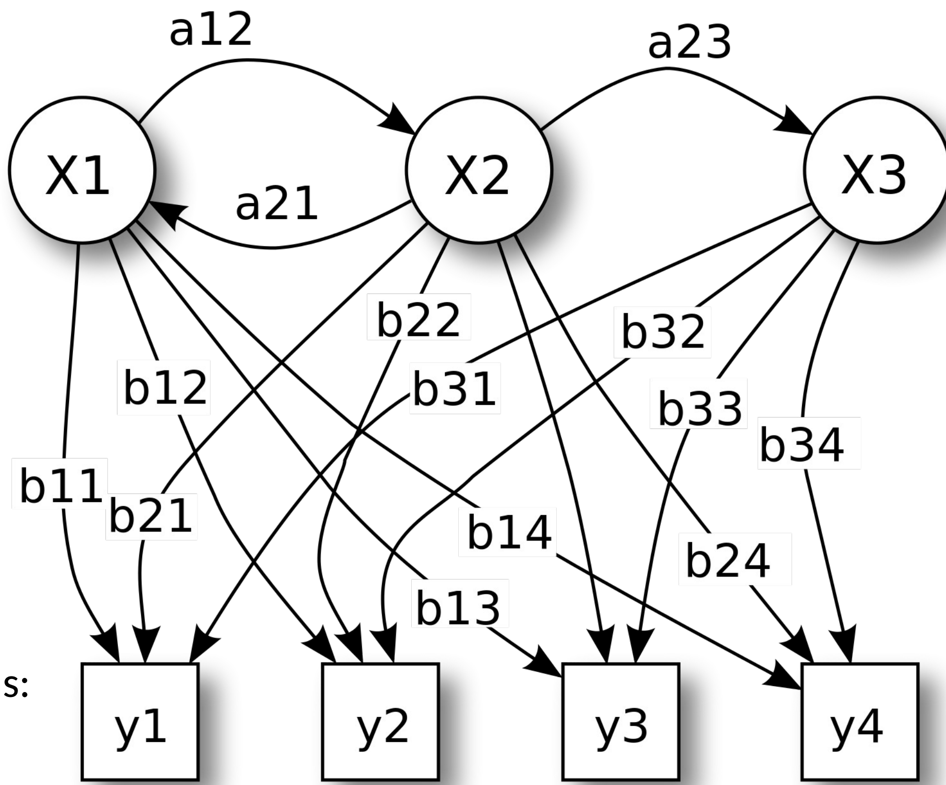
Observations:
words

X — states
y — possible observations
a — state transition probabilities
b — output probabilities

J&M section 8.4

HIDDEN MARKOV MODEL (HMM)

States:
labels



Observations:
words

X — states
y — possible observations
a — state transition probabilities
b — output probabilities

$$\hat{t}_{1:n} = \underset{t_1\ldots t_n}{\operatorname{argmax}} P(t_1\ldots t_n|w_1\ldots w_n) \approx \underset{t_1\ldots t_n}{\operatorname{argmax}} \prod_{i=1}^{n} \overbrace{P(w_i|t_i)}^{\text{emission}} \overbrace{P(t_i|t_{i-1})}^{\text{transition}} \tag{8.17}$$

TRAINING HMMS

- In HMM tagging, the probabilities are estimated by counting on a labelled training corpus (remember: Naïve Bayes from lecture 4)
- Task of determining the hidden variables sequence corresponding to the sequence of observations is called decoding.

$$\hat{t}_{1:n} = \underset{t_1 \ldots t_n}{\operatorname{argmax}} P(t_1 \ldots t_n | w_1 \ldots w_n) \approx \underset{t_1 \ldots t_n}{\operatorname{argmax}} \prod_{i=1}^{n} \overbrace{P(w_i|t_i)}^{\text{emission}} \overbrace{P(t_i|t_{i-1})}^{\text{transition}} \qquad (8.17)$$

TRAINING HMMS

- The decoding algorithm for HMMs is the Viterbi algorithm
- Viterbi: dynamic programming (remember: minimum edit distance algorithm from lecture 2)
  - Given an observation sequence, return the state path through the HMM that assigns maximum likelihood to the observation sequence
  - To compute the Viterbi probability at time t we use:

$v_{t-1}(i)$	the **previous Viterbi path probability** from the previous time step
a_{ij}	the **transition probability** from previous state q_i to current state q_j
$b_j(o_t)$	the **state observation likelihood** of the observation symbol o_t given the current state j

- J&M section 8.4.5

METHODS FOR NER

- Methods:
  - Feature-based
  - Neural-network-based

FEATURE-BASED NER

FEATURE-BASED NER

- Supervised learning:
  - Each word represented by a feature vector with information about the word and its context
  - x_i is the word in position i
    - Create a feature vector for x_i, describing x_i and its context

  - Training data needed: IOB-labeled texts

Steve	Jobs	was	a	co-founder	of	Apple	Inc.
B-PER	I-PER	O	O	O	O	B-ORG	I-ORG

FEATURE-BASED NER

- Supervised learning:
  - Each word represented by a feature vector with information about the word and its context
  - x_i is the word in position i
    - Create a feature vector for x_i, describing x_i and its context

- What features would you use for NER in the general domain (person names, place names, organizations, dates)?

FEATURE-BASED NER

- Commonly used features for sequence labelling NER:

identity of w_i, identity of neighboring words
embeddings for w_i, embeddings for neighboring words
part of speech of w_i, part of speech of neighboring words
presence of w_i in a **gazetteer**
w_i contains a particular prefix (from all prefixes of length ≤ 4)
w_i contains a particular suffix (from all suffixes of length ≤ 4)
word shape of w_i, word shape of neighboring words
short word shape of w_i, short word shape of neighboring words
gazetteer features

Figure 8.15 Typical features for a feature-based NER system.

SIDE STEP: PART-OF-SPEECH TAGGING

- Part-of-speech (POS) = ‘category of words that have similar grammatical properties’
  - noun, verb, adjective, adverb
  - pronoun, preposition, conjunction, determiner
  - Example:

    The/DT grand/JJ jury/NN commented/VBD on/IN a/DT number/NN of/IN other/JJ topics/NNS ./.
- Why would the POS of a word be informative for NER?

FEATURE-BASED NER

- Commonly used features for sequence labelling NER:

identity of w_i, identity of neighboring words
embeddings for w_i, embeddings for neighboring words
part of speech of w_i, part of speech of neighboring words
presence of w_i in a **gazetteer**
w_i contains a particular prefix (from all prefixes of length ≤ 4)
w_i contains a particular suffix (from all suffixes of length ≤ 4)
word shape of w_i, word shape of neighboring words
short word shape of w_i, short word shape of neighboring words
gazetteer features

Figure 8.15 Typical features for a feature-based NER system.

FEATURE-BASED NER

- Use of lists:
  - A gazetteer is a list of (place) names
  - Name lists (common first and last person names)
- Word shape features are used to represent the abstract letter pattern of the word by mapping lower-case letters to 'x', upper-case to 'X', numbers to 'd', and retaining punctuation. Thus for example I.M.F would map to X.X.X. and DC10-30 would map to XXdd-dd

FEATURE-BASED NER

- Commonly used features for sequence labelling NER:

identity of w_i, identity of neighboring words
embeddings for w_i, embeddings for neighboring words
part of speech of w_i, part of speech of neighboring words
presence of w_i in a **gazetteer**
w_i contains a particular prefix (from all prefixes of length ≤ 4)
w_i contains a particular suffix (from all suffixes of length ≤ 4)
word shape of w_i, word shape of neighboring words
short word shape of w_i, short word shape of neighboring words
gazetteer features

Figure 8.15 Typical features for a feature-based NER system.

- NER task: label all words in the sentence using a sequential model

CONDITIONAL RANDOM FIELDS (CRF)

- A discriminative undirected probabilistic graphical model
- Can take rich representations of observations (feature vectors)
- Takes previous labels and context observations into account
- Optimizes the sequence as a whole. The probability of the best sequence is computed by the Viterbi algorithm

J&M section 8.5

CONDITIONAL RANDOM FIELDS (CRF)

Length of sequence x

Sum over all feature function

Weight for given feature function

$\bar{x}$ is a sequence of words

Feature Functions

$$P(\overline{y}|\overline{x};w) = \frac{exp(\sum_{i}\sum_{j} w_j f_j(y_{i-1}, y_i, \overline{x}, i))}{\sum_{y' \in Y} exp(\sum_{i}\sum_{j} w_j f_j(y'_{i-1}, y'_i, \overline{x}, i))}$$

$\bar{y}$ a corresponding sequence of n tags

Sum over all possible label sequence

Feature function can access all of observation

- http://www.davidsbatista.net/blog/2017/11/13/Conditional_Random_Fields/

CONDITIONAL RANDOM FIELDS (CRF)

Length of sequence x

Sum over all feature function

Weight for given feature function

$$P(\overline{y}|\overline{x};w) = \frac{exp(\sum_i \sum_j w_j f_j(y_{i-1}, y_i, \overline{x}, i))}{\sum_{y' \in Y} exp(\sum_i \sum_j w_j f_j(y'_{i-1}, y'_i, \overline{x}, i))}$$

Feature Functions

Sum over all possible label sequence

Feature function can access all of observation

Feature functions:

```
features = {
    'bias': 1.0,
    'word.lower()': word.lower(),
    'word[-3:]': word[-3:],
    'word[-2:]': word[-2:],
    'word.isupper()': word.isupper(),
    'word.istitle()': word.istitle(),
    'word.isdigit()': word.isdigit(),
    'postag': postag,
    'postag[:2]': postag[:2],
}
```

- Implementation of CRF in sklearn: https://sklearn-crfsuite.readthedocs.io/en/latest/

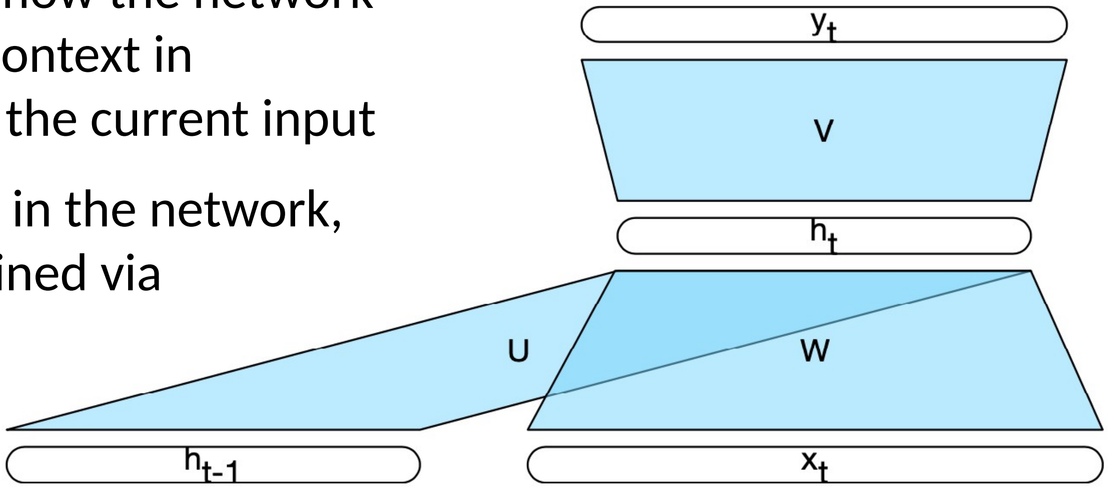
NEURAL MODELS FOR NER

NEURAL SEQUENCE MODELS

- Commonly used neural sequence model for NER: bi-LSTM
- LSTM = Long-short term memory
- Bi-LSTMs are Recurrent Neural Networks (RNNs)

RECURRENT NEURAL NETWORKS

- An RNN is any network that contains a cycle within its network connections.
- Similar to 'normal' feedforward network, but with 1 addition: a set of weights, *U*, that connect the hidden layer from the previous time step to the current hidden layer
  - These weights determine how the network should make use of past context in calculating the output for the current input
  - As with the other weights in the network, these connections are trained via backpropagation



BI-LSTMS

- Bidirectional neural model for NER: bi-LSTM
  - Bi-LSTM = Bidirectional Long Short-Term Memory
  - Word and character embeddings are computed for input word w_i and the context words
  - These are passed through a bidirectional LSTM, whose outputs are concatenated to produce a single output layer at position *i*
  - Simplest approach: direct pass to softmax layer to choose tag t_i

BI-LSTMS

- For NER the softmax approach is insufficient:
  - strong constraints for neighboring tokens needed (e.g., the tag I-PER must follow I-PER or B-PER)
  - Use CRF layer on top of the bi-LSTM output: biLSTM-CRF

- BiLSTM-CRF was the state of the art for NER for some years.

BI-LSTM-CRF FOR NER

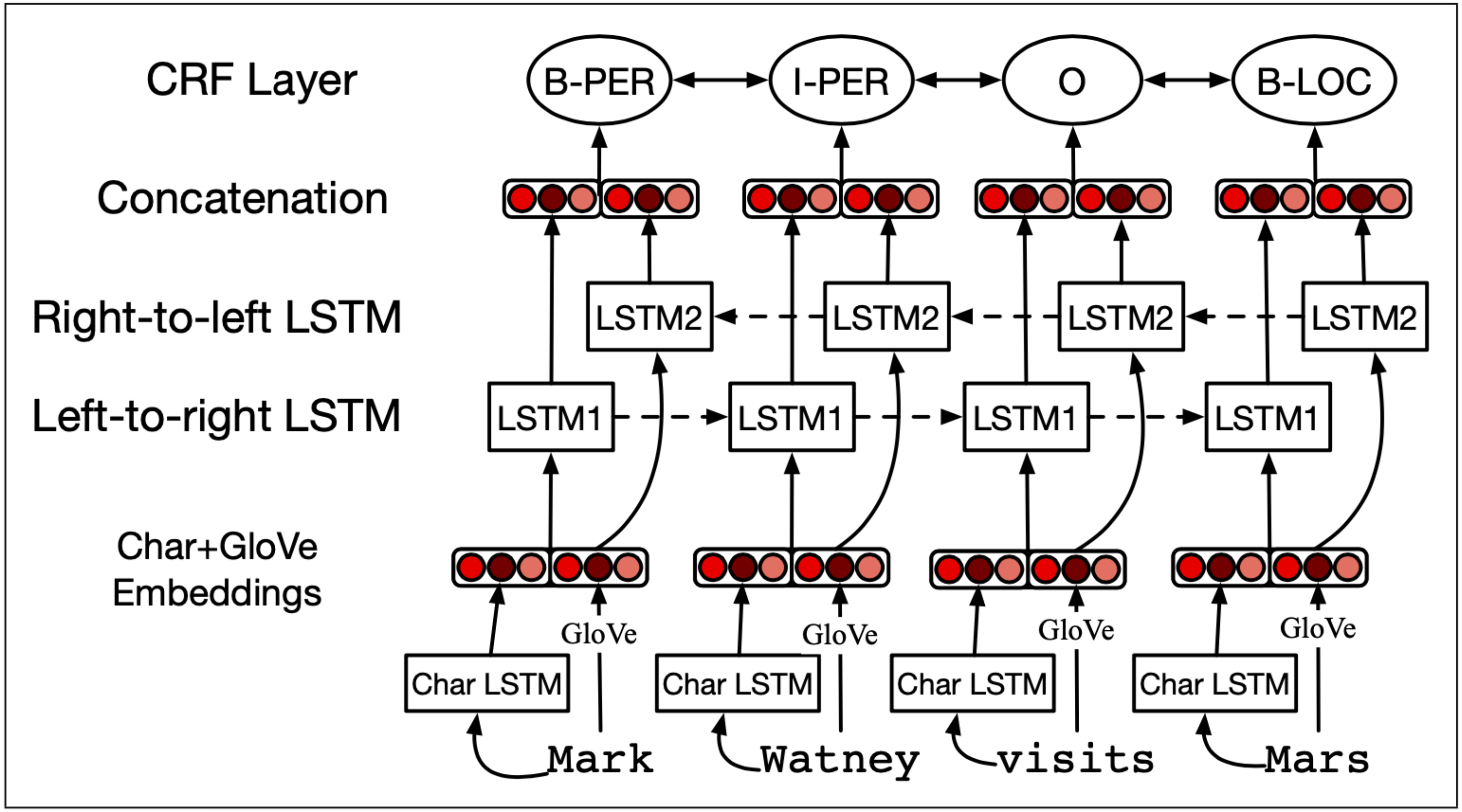


Figure 18.8 Putting it all together: character embeddings and words together in a bi-LSTM sequence model. After Lample et al. (2016).

TRANSFORMER MODELS

- Current state of the art for Named Entity Recognition: Transformer architectures
- In particular: BERT
- More about that next week

STATE OF THE ART FOR NER

- http://nlpprogress.com/english/named_entity_recognition.html
- Results on the CONLL-2003 benchmark:

Model	F1	Paper / Source	Code
ACE + document-context (Wang et al., 2021)	94.6	Automated Concatenation of Embeddings for Structured Prediction	Official
LUKE (Yamada et al., 2020)	94.3	LUKE: Deep Contextualized Entity Representations with Entity-aware Self-attention	Official
CL-KL (Wang et al., 2021)	93.85	Improving Named Entity Recognition by External Context Retrieving and Cooperative Learning	Official

RELATION EXTRACTION

J&M 17.1 AND 17.2

Universiteit Leiden

Suzan Verberne 2021

RELATION EXTRACTION

- Example text with named entities:

  Citing high fuel prices, [ORG United Airlines] said [TIME Friday] it has increased fares by [MONEY $6] per round trip on flights to some cities also served by lower-cost carriers. [ORG American Airlines], a unit of [ORG AMR Corp.], immediately matched the move, spokesman [PER Tim Wagner] said. [ORG United], a unit of [ORG UAL Corp.], said the increase took effect [TIME Thursday] and applies to most routes where it competes against discount carriers, such as [LOC Chicago] to [LOC Dallas] and [LOC Denver] to [LOC San Francisco].

- Relations:
  - *Tim Wagner* is a spokesman for *American Airlines*
  - *United* is a unit of *UAL Corp.*
  - etc

METHODS FOR RELATION EXTRACTION

1. Supervised learning (17.2.2)
2. Distant supervision (17.2.4)

- Option 1 is the most reliable. However, supervised learning requires labelled data. If labelled data is limited, we need option 2.

SUPERVISED RELATION EXTRACTION

- Assumptions:
  - Two entities, one relation
  - Relation is verbalized in one sentence
- Relation extraction as classification problem
  1. For each pair of entities in a sentence,
  2. determine whether or not they have a relationship
  3. and if they do, what the relation is

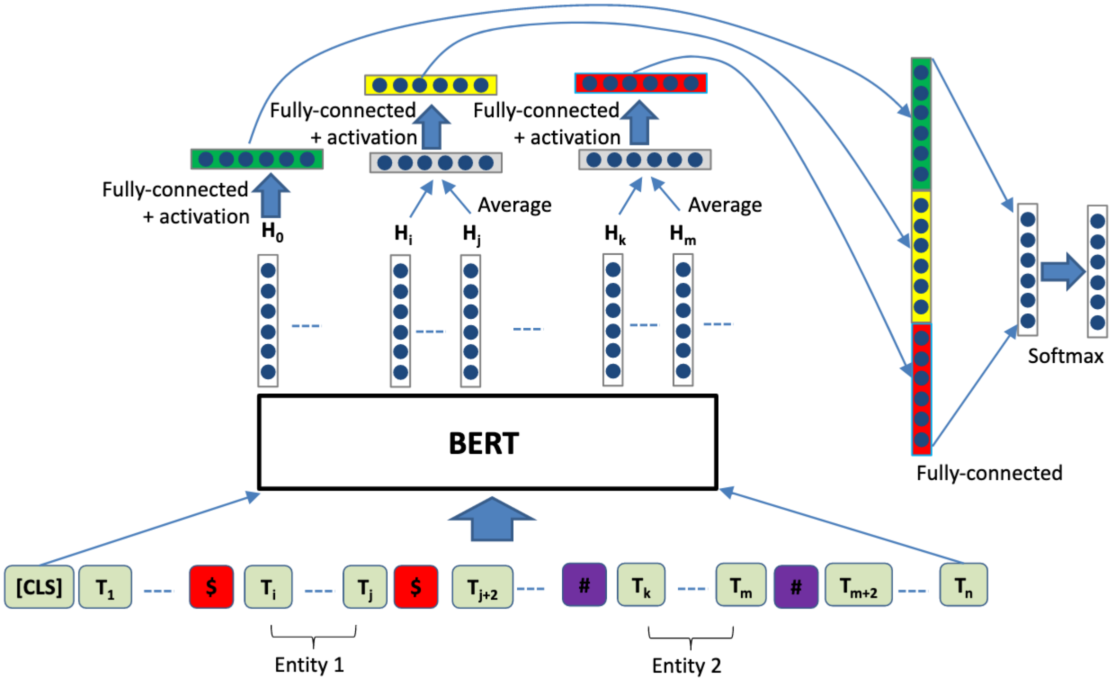
SUPERVISED RELATION EXTRACTION

- Feature-based approach
  - co-occurrence frequencies
  - entity features (words in the phrase, entity type e.g. person)
  - lexical contextual features (e.g. the word ‘founded’)
  - syntactic contextual features (e.g. SUBJ – ‘founded’ – OBJ)
  - background knowledge (e.g. clusters of entities from a large embeddings model)

SUPERVISED RELATION EXTRACTION

- Or neural architectures that do not require feature engineering

(R-BERT)



Wu, S., & He, Y. (2019, November). Enriching pre-trained language model with entity information for relation classification. In *Proceedings of the 28th ACM International Conference on Information and Knowledge Management* (pp. 2361-2364).

LIMITED LABELLED DATA

- Suppose we don't have labelled data for relation extraction, but we do have a knowledge base (e.g. DBpedia)
- How could you use the knowledge base to identify relations in the text and discover relations that are not yet in the knowledge base?
- Discuss with your neighbour

DISTANT SUPERVISION FOR RELATION EXTRACTION

1. Start with a large, manually created knowledge base (e.g. Freebase, DBpedia)

2. Find occurrences of pairs of related entities from the database in sentences
   - Assumption: If two entities participate in a relation, any sentence that contains these entities express that relation

3. Train a Relation Extraction classifier (supervised) on the found entities and their context

4. Apply the classifier to sentences with yet unconnected other entities in order to find new relations

DISTANT SUPERVISION FOR RELATION EXTRACTION

- The distant supervision paradigm is 12 years old:

  [PDF] Distant supervision for relation extraction without labeled data
  M Mintz, S Bills, R Snow, D Jurafsky - … of the Joint Conference of the 47th …, 2009 - aclweb.org
  Modern models of relation extraction for tasks like ACE are based on supervised learning of relations from small hand-labeled corpora. We investigate an alternative paradigm that does not require labeled corpora, avoiding the domain dependence of ACE-style algorithms, and allowing the use of corpora of any size. Our experiments use Freebase, a large semantic database of several thousand relations, to provide distant supervision. For each pair of entities that appears in some Freebase relation, we find all sentences containing those …
  Cited by 2665 Related articles All 27 versions

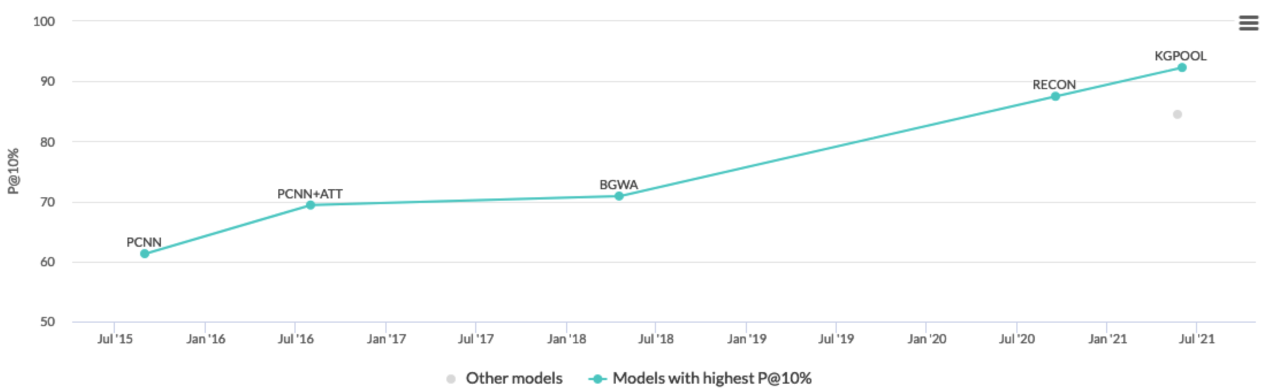
- But still applied in domains with limited labelled data

DISTANT SUPERVISION FOR RELATION EXTRACTION

Relationship Extraction (Distant Supervised) on New York Times Corpus

Leaderboard Dataset

View P@10% by Date



Filter: untagged

Edit Leaderboard

Rank	Model	P@10%	P@30%	AUC	Average Precision	Paper	Code	Result	Year	Tags
1	**KGPOOL**	92.3	86.7			KGPool: Dynamic Knowledge Graph Context Selection for Relation Extraction			2021	

Universiteit Leiden

https://paperswithcode.com/sota/relationship-extraction-distant-supervised-on

STATE OF THE ART FOR RELATION EXTRACTION

- http://nlpprogress.com/english/relationship_extraction.html
- https://paperswithcode.com/task/relation-extraction

End-to-End Models

Model	F1	Paper / Source	Code
BERT-based Models			
Matching-the-Blanks (Baldini Soares et al., 2019)	**89.5**	Matching the Blanks: Distributional Similarity for Relation Learning	
R-BERT (Wu et al. 2019)	**89.25**	Enriching Pre-trained Language Model with Entity Information for Relation Classification	mickeystroller's Reimplementation
CNN-based Models			
Multi-Attention CNN (Wang et al. 2016)	**88.0**	Relation Classification via Multi-Level Attention CNNs	lawlietAi's Reimplementation
Attention CNN (Huang and Y Shen, 2016)	84.3 85.9*	Attention-Based Convolutional Neural Network for Semantic Relation Extraction	

Universiteit Leiden

CONCLUSIONS

SUZAN VERBERNE 2021

Universiteit Leiden

HOMEWORK

- Read:
  - J&M chapter 8. Sequence Labeling for Parts of Speech and Named Entities
  - J&M chapter 17. Information Extraction

HOMEWORK

- Exercise week 6: named entity recognition with CRFsuite
  - Follow the tutorial on: https://sklearn-crfsuite.readthedocs.io/en/latest/tutorial.html
  - Make sure you understand what the steps mean
  - This is **not** a hand-in assignment. After next lecture, you will complete the 2nd hand-in assignment on this topic (deadline Nov 15)

- Next week we don't have a lecture

AFTER THIS LECTURE…

- You can describe the process of Named Entity Recognition (NER) as supervised sequence learning task using ‘IOB’ labels
- You can list a few commonly used features in NER
- You can explain MEMM and CRF for sequence labelling on a conceptual level
- You can explain recurrent neural networks for sequence labelling on a conceptual level
- You can describe distant supervision for extracting relations between two entities