Rich Context Competition

Allen Institute for Artificial Intelligence

Feb 15, 2019

Daniel King, Suchin Gururangan, Christine Betts, Iz Beltagy, Waleed Ammar, Madeleine van Zuylen

About us

- Allen Institute for Artificial Intelligence (https://allenai.org/)
    - Semantic Scholar (https://www.semanticscholar.org/)
    - AllenNLP (https://allennlp.org/)

Motivation

- Semantic Scholar augments papers with extracted and external content
    - e.g. extracted images, number of influential citations, github repositories related to the paper
- Dataset usage would be a useful addition to extracted content

Components

- Dataset Extraction
- Method Extraction
- Field of Research Prediction

Dataset Extraction

Dataset Extraction - Data

- ~10,000 datasets in a knowledge base

Dataset Extraction - Data

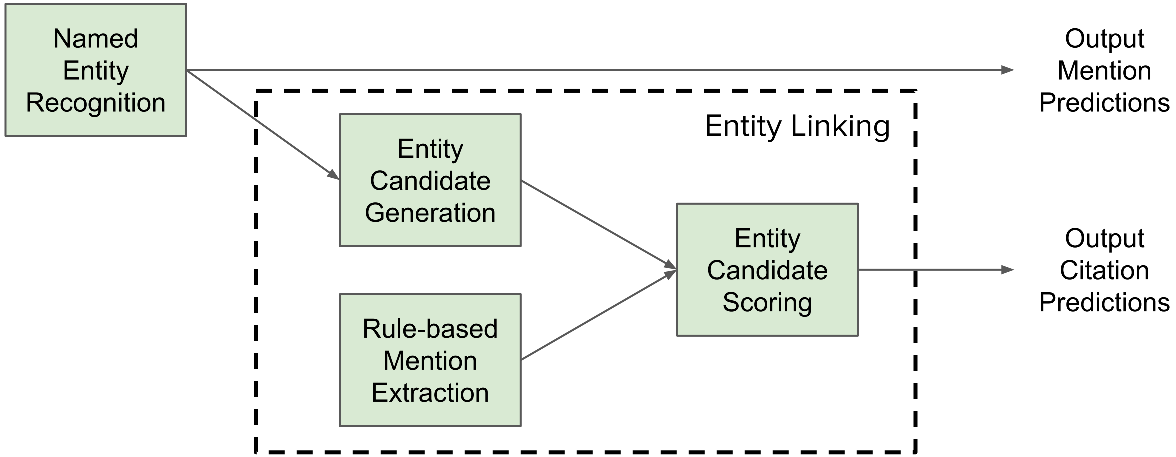
- ~10,000 datasets in a knowledge base
  - Many datasets share very similar names, and identical example mentions
    - e.g. Monitoring the Future: A Continuing Study of the Lifestyles and Values of Youth, 1980 and Monitoring the Future: A Continuing Study of the Lifestyles and Values of Youth, 1983 and Monitoring the Future: A Continuing Study of American Youth (12th-Grade Survey), 1996

Dataset Extraction - Data

- ~10,000 datasets in a knowledge base
  - Many datasets share very similar names, and identical example mentions
    - e.g. Monitoring the Future: A Continuing Study of the Lifestyles and Values of Youth, 1980 and Monitoring the Future: A Continuing Study of the Lifestyles and Values of Youth, 1983 and Monitoring the Future: A Continuing Study of American Youth (12th-Grade Survey), 1996
- 5,000 paper corpus with dataset usage labeled in them
  - ~10% of the datasets appear in the corpus
  - Example annotation: dataset X is referred to in paper Y as *American Community Survey*
  - The annotation does not say where in the paper the mention appears
  - Dataset usage labels contain an element of subjectivity

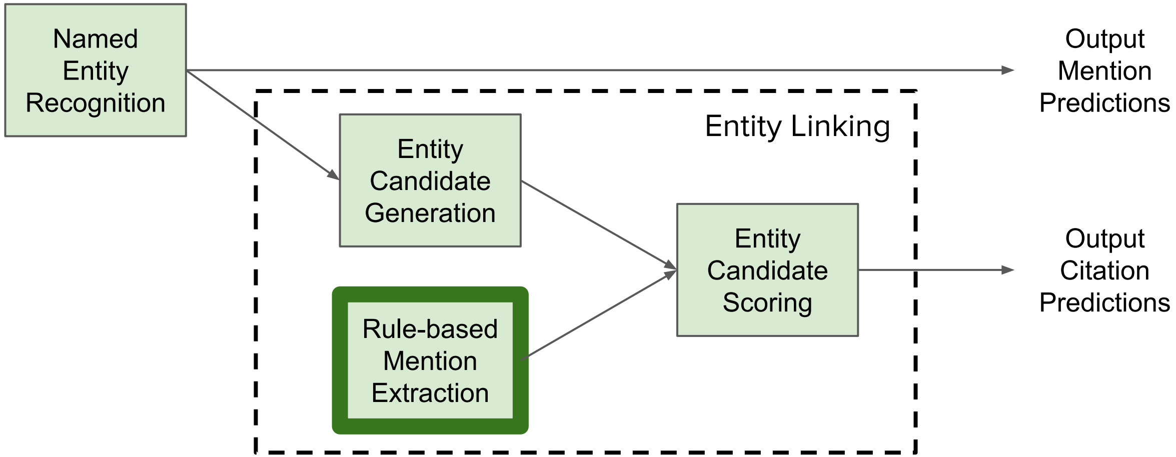
Dataset Extraction - Overview

- Task definition:
  - Input: a paper, knowledge base of datasets
  - Output: datasets used in that paper, both linked mentions and unlinked mentions
- Task fits into a common information extraction framework
  - Named entity recognition (NER)
  - Entity linking



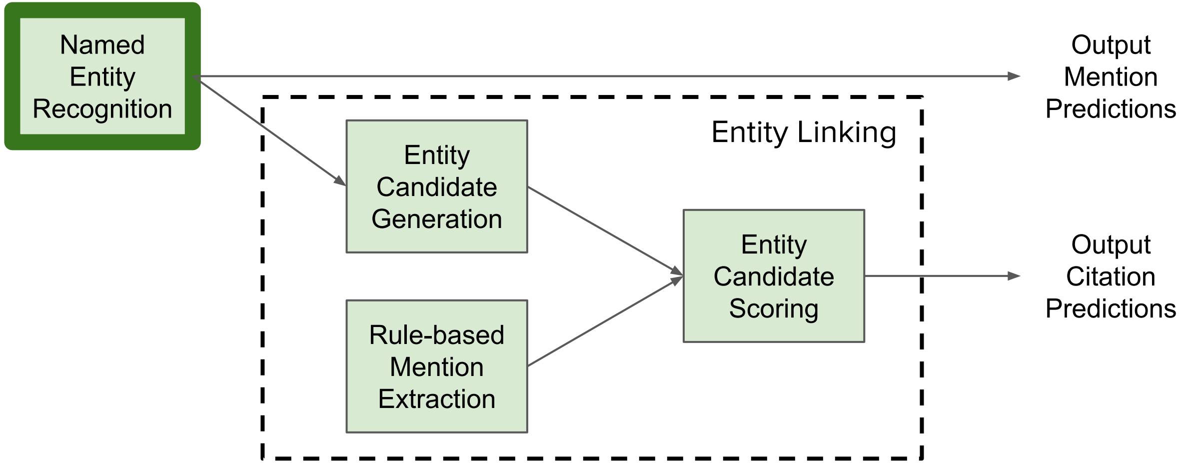
Dataset Extraction - Rule-based Mention Identification

- Lots of examples of datasets of interest provided in the knowledge base
- Regex search for these example mentions
- Built in candidates for entity linking



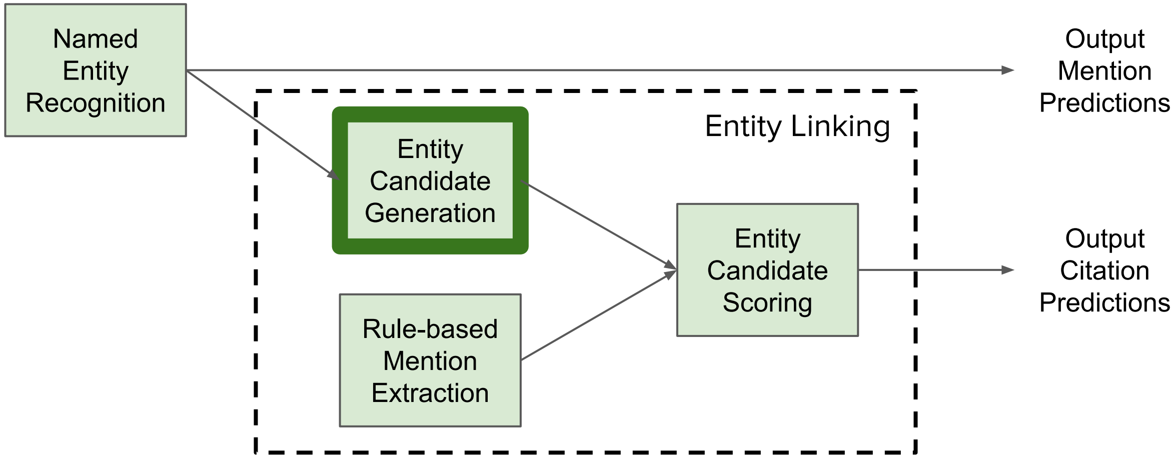
Dataset Extraction - Named Entity Recognition

- biLSTM with a CRF layer ([Deep contextualized word representations]), created using AllenNLP
  - Neural network model for sequence prediction
- Predicts textual mentions in a paper
- Trained based on noisy labeling using the provided knowledge base and corpus



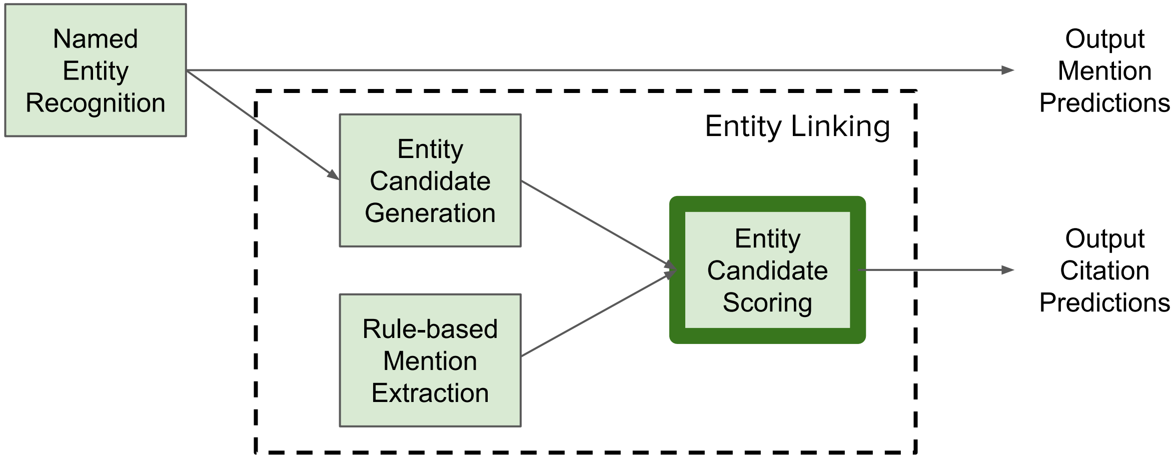
Dataset Extraction - Entity Candidate Generation

- Candidate datasets for each mention are generated by scoring all datasets based on TF-IDF weighted token overlap between the dataset title and the extracted mention text
  - High scoring pair: ‘Monitoring the Future 2006’ and ‘Monitoring the Future Survey from 2006’
  - Lower scoring pair: ‘American Community Survey 2009’ and ‘National Survey’



Dataset Extraction - Entity Candidate Scoring

- Gradient boosted trees classifier
  - Input: a candidate (mention text, dataset) pair
  - Output: probability that the pair is a correct extraction



Dataset Extraction - Results and Limitations

- Evaluation of just NER on constructed test set
    - F1: 0.46, precision: 0.51, recall: 0.42

Dataset Extraction - Results and Limitations

- Evaluation of just NER on constructed test set
  - F1: 0.46, precision: 0.51, recall: 0.42
- End to end evaluation

Dataset Extraction - Results and Limitations

- Evaluation of just NER on constructed test set
  - F1: 0.46, precision: 0.51, recall: 0.42
- End to end evaluation
  - Dev set provided in phase 1
    - Candidate generation - F1: 0.06, precision: 0.03, recall: 0.88
    - Candidate scoring - F1: 0.59, precision: 0.56, recall: 0.62

Dataset Extraction - Results and Limitations

- Evaluation of just NER on constructed test set
  - F1: 0.46, precision: 0.51, recall: 0.42
- End to end evaluation
  - Dev set provided in phase 1
    - Candidate generation - F1: 0.06, precision: 0.03, recall: 0.88
    - Candidate scoring - F1: 0.59, precision: 0.56, recall: 0.62
  - Subset of phase 1 holdout set
    - Candidate generation - F1: 0.04, precision: 0.02, recall: 0.60
    - Candidate scoring - F1: 0.27, precision: 0.36, recall: 0.22

Dataset Extraction - Results and Limitations (cont)

- Performs much better on datasets it has seen examples of
    - Due to rule based mention identification and noisy training data for named entity recognition

Dataset Extraction - Results and Limitations (cont)

- Performs much better on datasets it has seen examples of
  - Due to rule based mention identification and noisy training data for named entity recognition
- NER model tends to recognize acronyms, mentions it has seen before, and noun phrases containing words like ‘study’ or ‘survey’

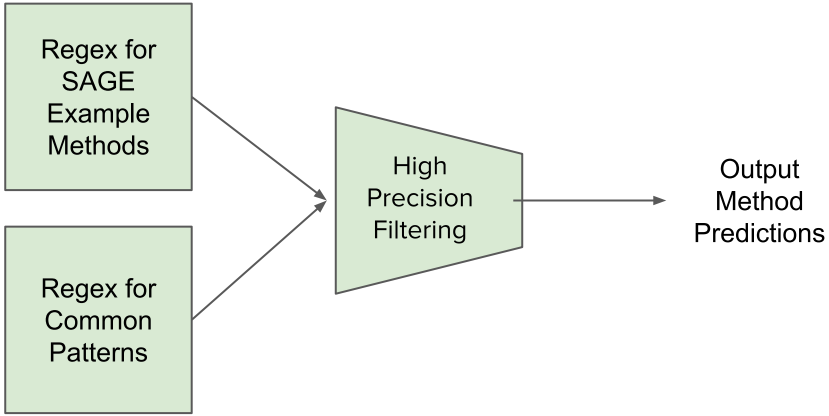
Dataset Extraction - Results and Limitations (cont)

- Performs much better on datasets it has seen examples of
  - Due to rule based mention identification and noisy training data for named entity recognition
- NER model tends to recognize acronyms, mentions it has seen before, and noun phrases containing words like ‘study’ or ‘survey’
- Each dataset candidate is scored independently

Method Extraction

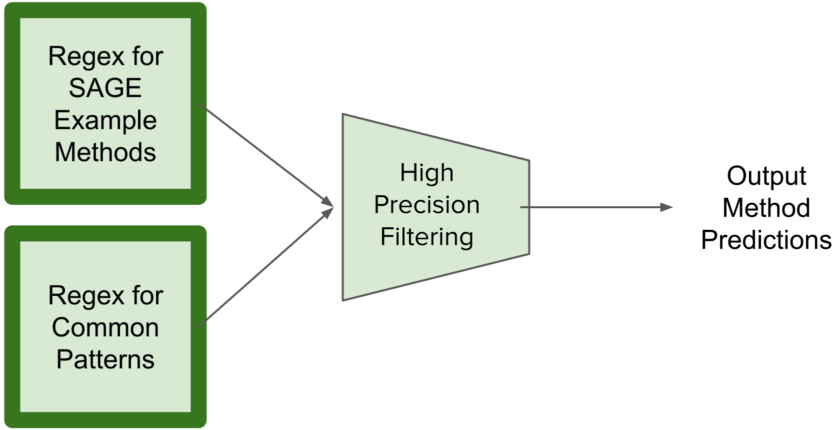
Method Extraction - Overview

- Task definition:
  - Input: a paper
  - Output: methods used in the paper
- Examine the SAGE ontology and some papers to understand what a method is
- Define regular expression to search for candidate methods
- Filter these candidate methods based on hand engineered rules



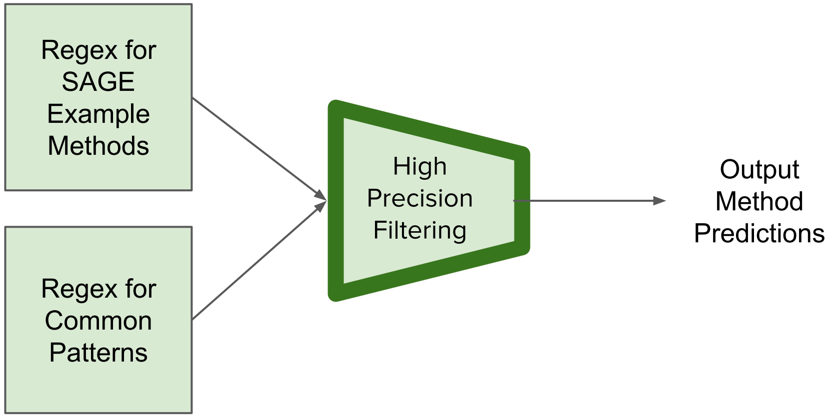
Method Extraction - Regular Expressions

- Search for phrases in the example ontology
  - Example ontology provided by the organizers, from SAGE
  - e.g. bivariate regression, longitudinal analysis
- Search for phrases ending in common ‘method’ words
  - e.g. Analysis, Theory, Model



Method Extraction - Filtering Candidate Methods

- Filter candidates based on hand-designed rules
  - e.g. capitalization, sentence length, word length
- Score candidates based on term frequency in a background corpus



Method Extraction - Results and Limitations

- From manual examination on a subset of the provided dev set
  - Precision: ~95%
  - Yield: ~1.5 methods per paper

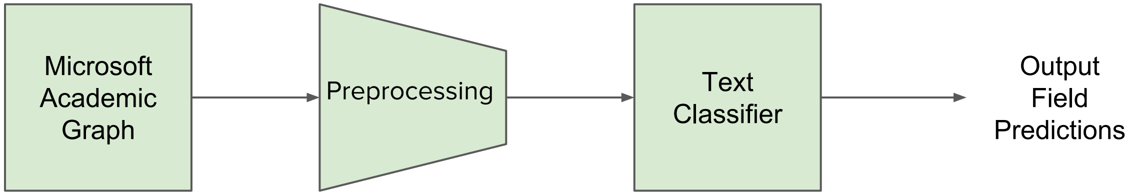
Method Extraction - Results and Limitations

- From manual examination on a subset of the provided dev set
  - Precision: ~95%
  - Yield: ~1.5 methods per paper
- Cannot find methods that don’t match SAGE ontology or our patterns
- Unclear exactly what a successful method extraction looks like

Field of Research Prediction

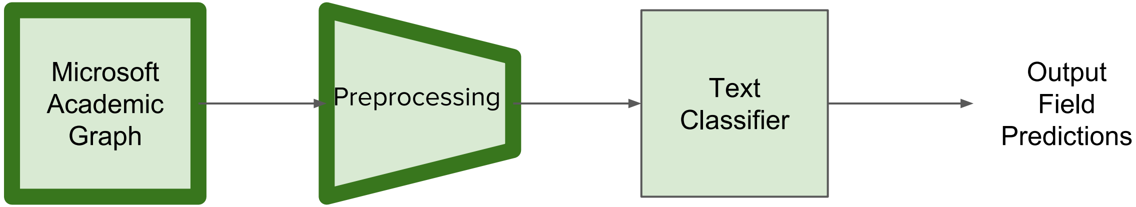
Field Prediction - Overview

- Task definition:
  - Input: a paper
  - Output: field of research of the paper
- Train a text classifier to predict field of study from publication title
- Trained on labeled data acquired from the Microsoft Academic Graph (https://academic.microsoft.com/)



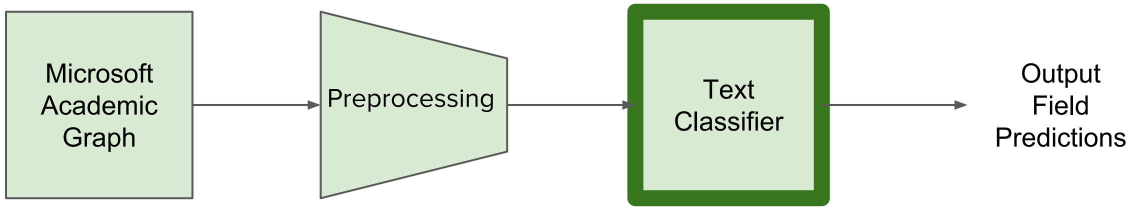
Field Prediction - Microsoft Academic Graph

- Microsoft Academic Graph (MAG) contains a hierarchy of topics, or fields of studies
  - [A Web-scale system for scientific knowledge exploration]
- Papers in MAG are tagged with these topics
- Training data from the top two hierarchy levels, filtered to fields of interest
  - L0 is the top-level (e.g. economics, medicine)
  - L1 is the second-level (e.g. econometrics, intensive care medicine)



Field Prediction - Text Classifier

- biLSTM neural network using ELMo word embeddings, created using AllenNLP
- Text classifier:
  - Input: a paper title
  - Output: top-level (L0) and second-level (L1) fields of study
- Always output the L0 prediction
- Output the L1 prediction when the model's confidence is high



Field Prediction - Results and Limitations

- Results on held out set from MAG
    - L0: 84.4% accuracy
    - L1: 65.2% accuracy

Field Prediction - Results and Limitations

- Results on held out set from MAG
  - L0: 84.4% accuracy
  - L1: 65.2% accuracy
- Only makes use of the titles of papers when predicting field of study
  - We are only able to access titles of papers in MAG
- Limited to the fields of study that MAG identifies
  - Will not be able to discover or generate new fields

Future Directions

- Dataset extraction
  - Improve NER model
  - Label example mentions in the actual text, rather than just by extracting strings
  - Define more clearly the difference between dataset reference and dataset usage
  - Explore patterns for identifying longer, more descriptive dataset mentions

Future Directions

- Dataset extraction
  - Improve NER model
  - Label example mentions in the actual text, rather than just by extracting strings
  - Define more clearly the difference between dataset reference and dataset usage
  - Explore patterns for identifying longer, more descriptive dataset mentions
- Method extraction
  - Further exploration of an open information extraction approach to detecting methods
  - Collecting more examples of what a successful method extraction looks like

Future Directions

- Dataset extraction
  - Improve NER model
  - Label example mentions in the actual text, rather than just by extracting strings
  - Define more clearly the difference between dataset reference and dataset usage
  - Explore patterns for identifying longer, more descriptive dataset mentions
- Method extraction
  - Further exploration of an open information extraction approach to detecting methods
  - Collecting more examples of what a successful method extraction looks like
- Field of research classification
  - Incorporate more than the title into prediction

Questions?

Thank you:

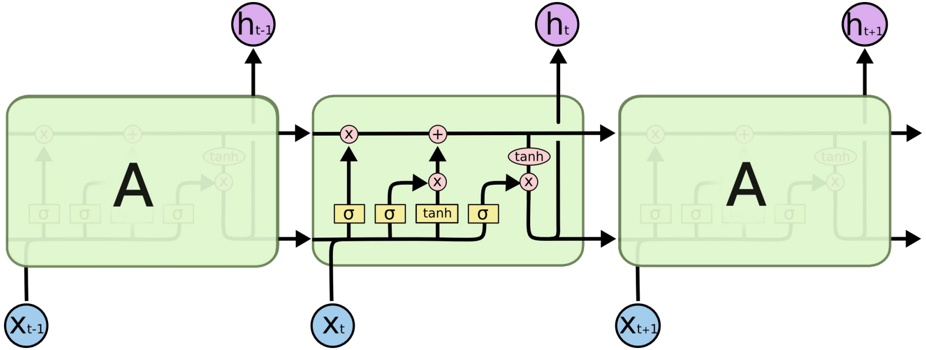
Suchin Gururangan, Waleed Ammar, Christine Betts, Iz Beltagy, Madeleine van Zuylen

All the organizers and sponsors of the competition

Appendix: hand engineered features

- Prior probability of entity
- Prior probability of entity given mention
- Prior probability of mention given entity
- Year matching between mention context and dataset title
- Mention length
- Mention sentence length
- Whether the mention is an acronym
- Approximate what section the mention is in
- Overlap between mention context and keywords + dataset subjects
- Score from the TFIDF weighted token overlap stage

Appendix: LSTM + CRF



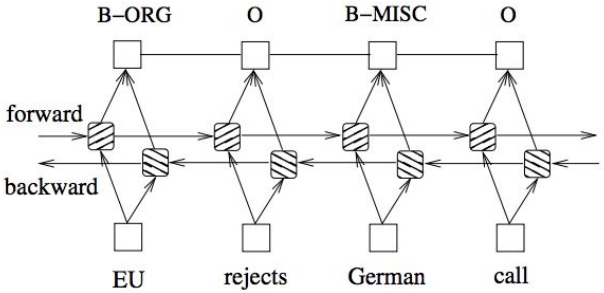


Figure 7: A BI-LSTM-CRF model.

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

From Bidirectional LSTM-CRF Models for Sequence Tagging

Appendix: NER config

```
    "percent_negatives": 50,
    "cutoff_sentence_length": 40

    "constrain_crf_decoding": true,
    "dropout": 0.75,
    "include_start_end_transitions": false,

"text_field_embedder": {
      "token_embedders": {
        "tokens": {
            "type": "embedding",
            "embedding_dim": 50,
            "pretrained_file": "/glove.6B.50d.txt",
            "trainable": true
        },
        "token_characters": {
            "type": "character_encoding",
            "embedding": {
                "embedding_dim": 16
            },
            "encoder": {
                "type": "cnn",
                "embedding_dim": 16,
                "num_filters": 64,
                "ngram_filter_sizes": [2, 3, 4],
                "conv_layer_activation": "relu",
                "output_dim": 128
            }
          }
      },
    },
    "encoder": {
        "type": "lstm",
        "input_size": 178,
        "hidden_size": 200,
        "num_layers": 2,
        "dropout": 0.5,
        "bidirectional": true
    },
    "initializer": [
      [".*tag_projection_layer.*weight", {"type":
"xavier_uniform"}],
      [".*tag_projection_layer.*bias", {"type": "zero"}],
      [".*feedforward.*weight", {"type": "xavier_uniform"}],
      [".*feedforward.*bias", {"type": "zero"}],
      [".*weight_ih.*", {"type": "xavier_uniform"}],
      [".*weight_hh.*", {"type": "orthogonal"}],
      [".*bias_ih.*", {"type": "zero"}],
      [".*bias_hh.*", {"type": "lstm_hidden_bias"}]
    ]
  },
  "iterator": {
    "type": "bucket",
    "batch_size": 16,
    "sorting_keys": [["tokens", "num_tokens"]]
  },
  "trainer": {
    "optimizer": {
        "type": "adam",
        "lr": 0.001
    }
```

Appendix: Test classifier config

```
  "token_indexers": {
      "tokens": {
          "type": "single_id",
          "namespace": "tokens",
          "lowercase_tokens": true,
      },
      "elmo": {
          "type": "elmo_characters",
          }
  },
  "sequence_length": 400
},
"model": {
  "type": "classifier",
  "text_field_embedder": {
      "token_embedders": {
          "tokens": {
              "type": "embedding",
              "embedding_dim": 300,
              "trainable": true,
          },
          "elmo": {
              "type": "elmo_token_embedder",
              "do_layer_norm": false,
              "dropout": 0.2
          }
      },
  },
```

```
  "encoder": {
    "type": "lstm",
    "num_layers": 2,
    "bidirectional": true,
    "input_size": 1324,
    "hidden_size": 128,
  },
  "aggregations": ["maxpool", "final_state"],
  "output_feedforward": {
      "input_dim": 512,
      "num_layers": 1,
      "hidden_dims": 128,
      "activations": "relu",
      "dropout": 0.5
  },
  "classification_layer": {
      "input_dim": 128,
      "num_layers": 1,
      "hidden_dims": 32,
      "activations": "linear"
  },
  "initializer": [
      [".*linear_layers.*weight", {"type": "xavier_uniform"}],
      [".*linear_layers.*bias", {"type": "zero"}],
      [".*weight_ih.*", {"type": "xavier_uniform"}],
      [".*weight_hh.*", {"type": "orthogonal"}],
      [".*bias_ih.*", {"type": "zero"}],
      [".*bias_hh.*", {"type": "lstm_hidden_bias"}]
  ]
},
```

```
"iterator": {
      "type": "bucket",
      "sorting_keys": [["tokens", "num_tokens"]],
      "batch_size": 32
  },
  "trainer": {
      "optimizer": {
          "type": "adam",
          "lr": 0.0004
      },
      "validation_metric": "+accuracy",
      "num_serialized_models_to_keep": 2,
      "num_epochs": 75,
      "grad_norm": 10.0,
      "patience": 5,
      "cuda_device": 0,
      "learning_rate_scheduler": {
          "type": "reduce_on_plateau",
          "factor": 0.5,
          "mode": "max",
          "patience": 0
      }
  }
}
```