CliNER: A Lightweight Tool for Clinical Named Entity Recognition

William Boag[§], Kevin Wacome[§], Tristan Naumann[¥], Anna Rumshisky[§]

[§]Text Machine Lab for Natural Language Processing, UMass Lowell

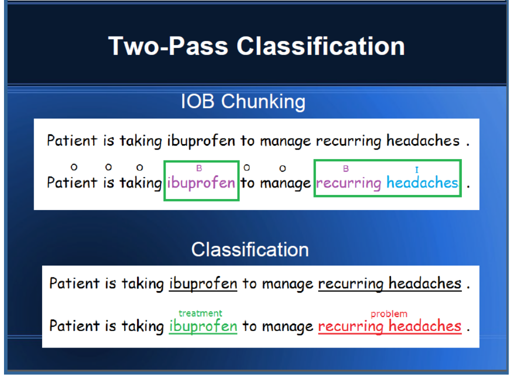
[¥]Clinical Decision-Making Group, MIT CSAIL

Named Entity Recognition

- Named Entity Recognition (NER) in the clinical domain aims to identify clinically relevant concepts in the provider narrative text of electronic medical records (EMR).
- Such concepts as diseases/disorders, treatments/medications, and tests have been the focus of clinical NER community challenges, such as 2010 i2b2/VA NLP challenge, 2013 CLEF-eHEALTH challenge, SemEval 2015 Task 14.

Why another NER system?

- Despite recent advances in clinical NER state-of-the-art, no lightweight, easy to set up, open-source implementation is available to date.
- Shared tasks helped to identify best-performing methods, but the workshop format does not allow or encourage teams to develop fully functioning, user-friendly systems.
- Systems developed during the challenges or published subsequently are never released for public use and frequently are put together haphazardly from opportunistically developed code components.
- On the other hand, the systems developed outside of the shared task paradigm tend to be heavy aggregations of multiple components that require extensive set up and configuration, presenting a significant barrier to initial use.



Two-Pass Classification

IOB Chunking

Patient is taking ibuprofen to manage recurring headaches .

Patient is taking ibuprofen to manage recurring headaches .

Classification

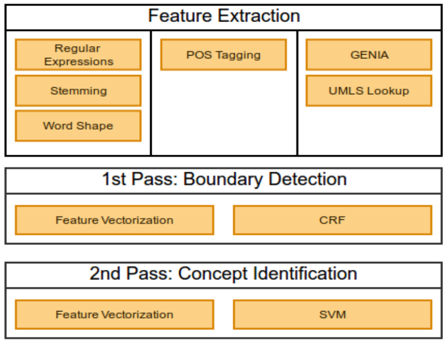
Patient is taking ibuprofen to manage recurring headaches .

Patient is taking ibuprofen to manage recurring headaches .

What is CliNER?

- Clinical Named Entity Recognition system (CliNER) is an open-source natural language processing system for named entity recognition in clinical text of electronic health records. CliNER is designed to follow best practices in clinical concept extraction.
- CliNER is implemented as a two-pass machine learning system for named entity recognition, currently using a Conditional Random Fields (CRF) classifier to establish concept boundaries and a Support Vector Machine (SVM) classifier to establish the type of concept.

System Architecture



Feature Extraction

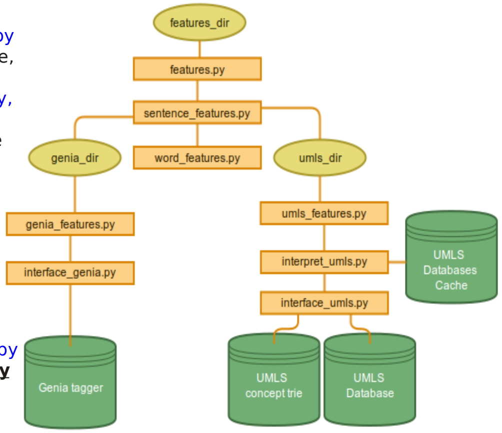
1st Pass: Boundary Detection

2nd Pass: Concept Identification

- Extensible, easy-to-use architecture
- Free software: Apache v2.0 license
- Available on GitHub, see the project website: http://cliner.org
- Implemented in Python, using sklearn, CRFsuite, and LibSVM
- Support for multiple formats, currently supporting:
  - word offset-based format
  - inline XML
  - character offset-based format

- **Basic system functionality**
  - train.py, predict.py, evaluate.py
- Utilities: command-line interface, installation & config tools, etc.
  - helper.py, is_installed.py, cli.py, format.py, read_config.py
- **Data representation**: pipeline logic, storage for the two-pass implementation, etc.
  - note.py, model.py
- **Features**

  features.py,
  sentence_features.py,
  word_features.py
  genia: genia_features.py,
  interface_genia.py
  umls: umls_features.py, umls.py
- **Machine Learning: ML library wrappers**
  - sci.py, crf.py



Features

1. Concept boundary detection
   - General text features:
     - previous 3 unigrams, next 3 unigrams, current word's POS, unigram w/digits replaced by #, other word shape features, previous two tokens' features, following two tokens' features
   - Genia features: GENIA stem, GENIA POS, GENIA chunk-tag
   - UMLS features: UMLS CUI, UMLS semantic type
   - Prose and non-prose contexts processed separately
2. Concept Type Identification
   - Additional features: regular expressions for dates, test results, doctor abbreviations

Results on i2b2/VA 2010 Data

	Precision	Recall	F-measure
Problem	0.710	0.858	0.777
Treatment	0.834	0.752	0.791
Test	0.840	0.825	0.833
Micro-average	0.795	0.812	0.800

Current Updates (Feb 2015)

- Support for disjoint named entity spans:
  - 3rd Pass: merging pairs of non-contiguous spans using an SVM classifier
  - Added syntactic features: collapsed dependencies using Stanford dependency parser.
- Normalizing named entities to UMLS concepts with MetaMap output filtered on semantic type of the entity.
  - Queries normalized with LVG and a custom spell-checker.
- Miscellanea:
  - Support for character-offset formats.
  - Installation and dependency diagnostics for easier setup.

System Output: Trained on i2b2/VA 2010 Data

BRIEF HISTORY: The patient is an (XX)-year-old female with history of <problem>previous stroke</problem> ; <problem> hypertension </problem> ; <problem>COPD</problem> , stable ; <problem>renal carcinoma </problem> . <test> CT of the maxillofacial area</test> showed no <problem>facial bone fracture </problem> . <test> Echocardiogram </test> showed normal left ventricular function . She was set up with a skilled nursing facility , where she was to be given <treatment>daily physical therapy</treatment> .