NER and Information Extraction

ZHENYA ANTIĆ
PRACTICAL LINGUISTICS, INC.
JULY 17, 2019
Overview

- Definitions
- Extracting named entities: names, locations, organizations
- Extracting concepts: products/services and matching them to knowledge base
- Topic modeling
- Relation extraction
- Turkish specific problems and solutions
- Additional notes
- References
Information extraction: extracting structured information from unstructured text, including entities and relations between them, sometimes also connecting to an existing knowledge base.

NER, named entity recognition: subtask of information extraction, extracting proper names and identifying their classes, such as people, locations, organizations, etc.

Similar to NER: concept extraction and linking. Extracting entities relevant to a particular domain (such as banking products and services).
Identifying named entities:

Turkey remembered prolific writer and poet Rifat Ilgaz on July 7, 26 years since he passed away.

Classifying named entities:

Turkey remembered prolific writer and poet Rifat Ilgaz on July 7, 26

LOCATION                      PERSON                   DATE
years since he passed away.

Use labeling schemes to indicate NE start, NE end, word inside NE, word outside NE, a singleton NE, or BIO (beginning, inside and end)
Preprocessing

- Tokenization
  - Splitting into sentences and words
- Lemmatization
  - Grouping together the inflected forms of the same word (such as “went” and “go”)
- Tagging the words with the part of speech (POS tagging)
NER techniques: rule-based

- High precision, low recall
- Labor intensive
- Need linguistic knowledge of the language
- Gazetteers can be useful when combined with other techniques
NER techniques: statistical machine learning

- Different ML classifiers can be used: HMM, Decision Trees, SVM, CRF
- Conditional Random Fields
  - Discriminative classifier (learns the boundary between classes)
  - Classifier where context is taken into account (one of the input features is the previous element’s label), hence best suited for such a task
- Introduction to CRF: https://medium.com/ml2vec/overview-of-conditional-random-fields-68a2a20fa541
- Turkish NER using CRF: https://web.itu.edu.tr/gulsenc/papers/NERsubmittedColing.pdf
NER techniques: clustering

- Unsupervised
- Several different clustering methods
- K nearest neighbors
  - Represent each data point as a vector of features, all are points in a hyperplane
  - Label is assigned by a majority vote of the points nearest to the given point (based on Euclidian distance)
- Some features: POS, word, upper/lower case, digits, previous word POS, previous word label, etc.
NER techniques: word embeddings

- Word embeddings: vector representations of words
- Obtained using a neural network that tries to predict words from the words that surround it (context)
- Generating embeddings is easy using a corpus using existing Python libraries (word2vec)
- Lots of NER systems use word embeddings as input features
- Semi-supervised NER on Turkish Twitter data: https://arxiv.org/pdf/1810.08732.pdf
NER techniques: deep learning

- Advantages of deep learning: minimal feature engineering
- Disadvantages: require lots of labeled data
- Many use word embeddings
- Another idea: use character level embeddings
  - Saves sub-word information, such as prefixes and suffixes (important for an agglutinative language such as Turkish)
  - Language-independent
- Usually, bidirectional LSTM networks are used
  - Long distance dependencies
NER techniques: deep learning

- A hybrid model combines LSTM and CRF ([https://arxiv.org/pdf/1709.09686.pdf](https://arxiv.org/pdf/1709.09686.pdf), Russian, approach could be similar to Turkish)

- State-of-the-art: character and word embeddings fed into a bidirectional LSTM and then into CRF
Three different IE tasks:

- Terminology Extraction: terminology used in a domain/corpus
- Keyphrase Extraction: extracting important phrases for a document
- Topic Modeling: cluster related keywords into higher level topics

Usually, later we either:

- Connect the extracted entities to a knowledge base
- Or, extend the knowledge base
Concept extraction

- Extraction: similar to NER
- Differences:
  - Capitalization is less useful
  - More syntactic processing is needed, since entities are more complex (
    "[inner planets] of the [solar system]"
)
Finding candidate entities:
- Extracting n-grams up to a predefined length
- POS tagging + shallow syntactic parsing
  - For example, noun phrase (NP) extraction

Filtering candidates:
- Rule-based
- Statistical
Concept extraction

- Analyze two key properties:
  - Unithood: cohesiveness of the term referring to the concept ("mean squared error")
    - Compare the expected number of times the collocation would appear if the individual words were independent versus the actual number of times the collocation appears
    - Could also compare web search results ("mean squared error" versus mean AND squared AND error)
  - Termhood
    - Relevance of the term to the domain in question
    - TF-IDF is commonly used
Cluster and analyze thematically related terms (e.g., “carcinoma”, “malignant tumor”, “chemotherapy”)

Assign a topic label to clusters, potentially from a Knowledge Base

LSA/LDA are traditional methods
  - Drawback: work on individual terms, not multiword entities
  - There are no labels for the topics
  - Words are not semantically interpreted
Instead, could use a Knowledge Base to turn this into a labeling problem

Or, first link to KB, and then apply LSA/LDA

Or, three levels instead of two: words, concepts linked to KB and topics

Graph-based approaches
  - Calculate centrality as a metric

A variety of tools in the IE for semantic web survey paper
(http://www.semantic-web-journal.net/system/files/swj1744.pdf)
Linking entities to Knowledge Base

- Link entities and topics to Knowledge Base
- Usually using graph methods
- Requires a knowledge base that has entities and relations between entities
Extracting relations

- Relations can be
  - Binary ("Barak is married to Michelle")
  - N-ary ("Cecile gave Mary a book")
- Tools used for relation extraction
  - Syntactic parsing
  - Semantic frames (WordNet, VerbNet, FrameNet, etc.)
  - Distant supervision
    - Two or more entities with a known relation in a KB mentioned together in a sentence are likely to have the relation as well
    - Problems: several relations for a distinct set of entities, noisy output
    - Recently: distant supervision based on embeddings
Extracting relations

- Bi-LSTM: treating this as a sequence problem, where there are predicate and arguments labels ([https://gabrielstanovsky.github.io/assets/papers/naacl18long/paper.pdf](https://gabrielstanovsky.github.io/assets/papers/naacl18long/paper.pdf))

Relation clustering

- Combining similar relations (“is-married-to”, “is-spouse-of”)

- Approaches:
  - Using semantic sources such as WordNet
  - Consider sets of entity pairs that each pattern considers. If the sets for two patterns are nearly identical, the two patterns must be semantically similar
- High number of morphological forms
- Turkish names are also common words
- Free word order language
- Some language-independent solutions (such as using character-based embeddings)
- Using solutions similar to other morphologically rich languages
NYC NLP talks: Tim Moller from Lexalytics, Paul Tepper from Nuance

Lexalytics

- BERT: achieve the same F1 scores with 10 times less training data (but 10X the training time and 10X the model size)
- Compared to SVM

Nuance

- Conversational AI
- Leverage conversational data that already exists: construct conversation graphs to be used to make chatbots
Similar to word embeddings, such as word2vec

- Difference: include context, so embeddings vectors are different depending on context
  - “The man was accused of robbing a bank.”
  - “The man went fishing by the bank of the river.”
  - Different encodings for “bank”

- Now have multilingual models
  - Top 100 languages with the largest Wikipedias (which includes Turkish)
Summary

- Statistical learning and neural networks are the main approaches
- Graph-based approaches for linking to a knowledge base
- Word-based and character-based embeddings are being used more and more
  - Recently BERT
- NER: word and character embeddings into bidirectional LSTM with CRF
- Concept extraction: similar to NER; add some syntactic parsing
- Extracting relations: also bidirectional LSTM with word embeddings
Thank you
References: NER

References: Information Extraction

- Information Extraction (book, https://www.cis.uni-muenchen.de/~fraser/information_extraction_2018_lecture/sarawagi.pdf)
- Information Extraction meets the Semantic Web: A Survey (http://www.semantic-web-journal.net/system/files/swj1744.pdf)
- Supervised Open Information Extraction (https://gabrielstanovsky.github.io/assets/papers/naacl18long/paper.pdf)
References: concept and relation extraction and linking

References: Turkish


Empirical evaluation of compounds indexing for Turkish texts

Turkish Natural Language Processing, including a chapter on NER
(https://link.springer.com/book/10.1007/978-3-319-90165-7, paywall)

Developing a concept extraction system for Turkish
(https://www.cmpe.boun.edu.tr/~gungort/theses/Developing%20a%20Concept%20Extraction%20System%20for%20Turkish2.pdf)

Initial explorations on using CRFs for Turkish Named Entity Recognition
(https://web.itu.edu.tr/gulsenc/papers/NERsubmittedColing.pdf)
References: Turkish

- Semi-supervised NER on Turkish Twitter data: https://arxiv.org/pdf/1810.08732.pdf