ABBYY

Извлечение именованных сущностей (NER)

Краткий обзор

Иван Смуров, Ivan_S@abbyy.com

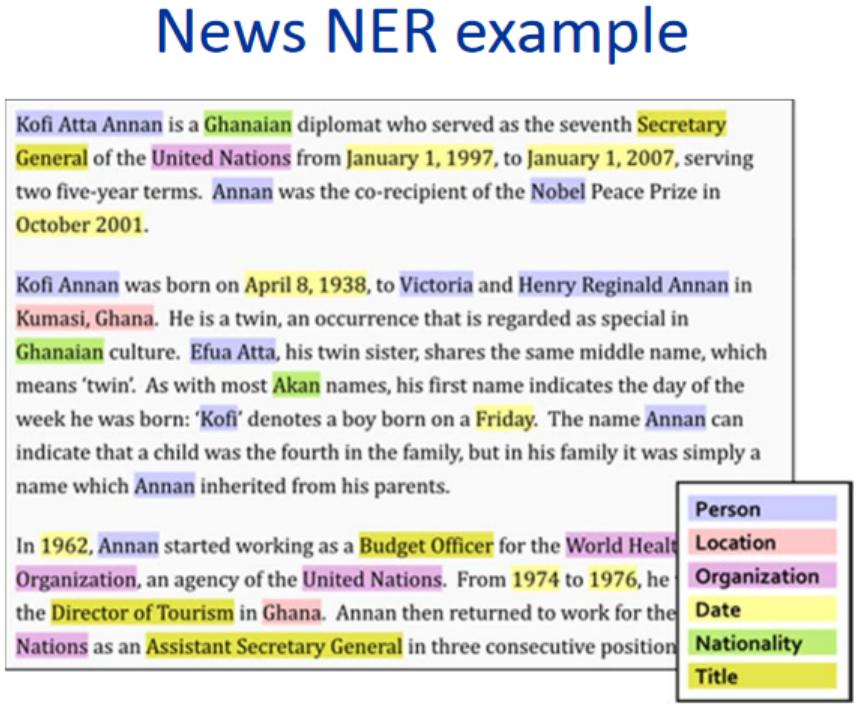
ABBYY

Содержание

- Формулировка задачи NER
- Классические решения задачи NER
- Нейросетевые решения задачи NER
- Извлечение именованных сущностей: от теории к практике

NER: постановка задачи

- Задача NER – выделение спанов сущностей в тексте
- В классической постановке (MUC-6, 1996 год) сущности – персоны, локации и организации
- В разных стандартных корпусах добавляются свои дополнительные типы сущностей - Misc, даты, денежные суммы и т. п



NER: зачем все это нужно?

ABBYY

- Сам по себе NER нужен не слишком часто (хотя встречаются и прямые практические применения NER: обычно приведение неструктурированных данных в более структурированный вид - текстов в таблицы и т. д.)
- Тем не менее, это шаг в сторону «понимания» текста – позволяет выделить в тексте важные зоны, собрать (или даже просто выделить в тексте) куски для дальнейшего анализа и т. п.
- Также, благодаря NER может улучшиться качество других задач NLP (сами сущности – надежные коллокации, выделение может помочь в разрешении местоименной анафоры и т. п.), можно улучшить качество понимания поисковых запросов и т. д.
- Постановка достаточно гибкая – можно подобрать нужный для конкретной задачи набор сущностей и научиться их выделять.

NER: в чем подвох?

ABBYY

- Задача не такая простая – есть сложности с омонимией (Вашингтон - персона или локация), необходимость учитывать глобальный контекст и знания о мире и т. п.
- Практически для любого набора сущностей возникают тонкие и пограничные случаи выделения – что является сущностью, как проводить границы спанов (Магазин Профессиональных Металлоискателей v. s. магазин зоотоваров Немо v. s. «Цветочек» - магазин лучших и самых любимых брендов по доступным ценам).
- Как результат, инструкция усложняется, требуется все более квалифицированная (а значит дорогая) разметка

NER: метрики и корпуса

ABBYY

- Для сравнения, как правило, используется строгая f-мера (сущность – true positive т. и т. т., к. границы спанов эталона и теста в точности совпадают)
- В силу дороговизны разметки общедоступных корпусов немного
- Для английского языка есть корпуса различных соревнований по NER – MUC, TAC, CoNLL. Везде, как правило, используются новостные тексты
- Золотой стандарт – CoNLL 2003 (~300k токенов, сущности – LOC, PER, ORG и Misc). SOTA f-мера ~ 0.93
- Для русского языка ситуация еще хуже: единственный доступный корпус Диалог 2016 очень маленький (~50к токенов) и со специфической разметкой

NER: сведение к задаче классификации

ABBYY

- BIOES-схема Иван Петрович Сидоров купил Google ->

  B-PER I-PER E-PER OUT S-ORG

- Проблемы с пересечением сущностей – МГУ им. Ломоносова
- Стандартный способ хранить данные – формат conll. Текст разбивается на предложения, предложения на токены. Строка соответствует одному токену. В колонках необходимая для анализа информация про токен (словоформа, POS-таг, метка и т д)

```
1   Статус   статус   S     S,муж,неод=(вин,ед|им,ед)    OUT
1   заказа   заказ    S     S,муж,неод=род,ед    B-ORDER
1   номер    номер    S     S,муж,неод=(вин,ед|им,ед)    I-ORDER
1   25738    25738    NUMBER   UNDEF    E-ORDER
1   от   от   PR   PR=  OUT
1   13.02    13.02    MAINLY_NUMERIC   UNDEF    OUT
1   .    .    PUNCT    PUNCT    OUT
1   2017     2017     NUMBER   UNDEF    OUT
```

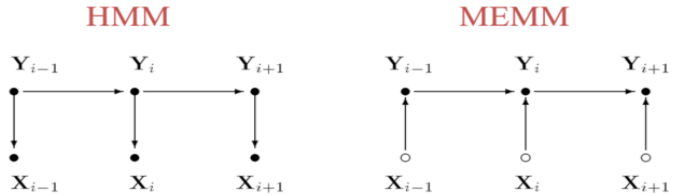
NER: дела давно минувших дней

ABBYY

- Строго говоря, задачу NER можно решать и без машинного обучения. Rule-based системы для отдельных сущностей (особенно числовые – даты, денежные суммы и т п) или корпусов могут дать неплохой результат
- Тем не менее, до конца 2000-х, SOTA-результаты показывали системы на основе классических методов машинного обучения(HMM, MEMM, SVM, CRF, random forest, их комбинации)
- В качестве признаков обычно использовалась словоформа, POS-таги, морфология (префиксы, суффиксы) а также признаки о наличии в токене спецсимволов и внешнем виде токена (капитализация, наличие пунктуаторов). Самый общий из последней категории – шаблон капитализации (iPhone6 -> aAaa1)
- Для улучшения качества активно используются газетиры(словари сущностей)
- Неплохой обзор классических методов для решения NER в *Nadeau and Sekine (2007), A survey of Named Entity Recognition and Classification*

Label bias problem

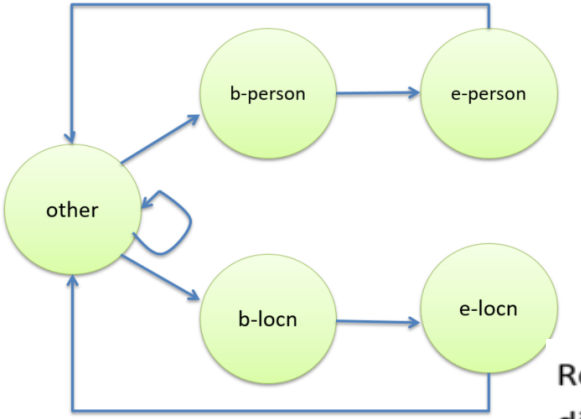
- Метки в схеме IOBES завсият друг от друга. Напр. метка I-Per может быть только после метки B-Per или I-Per .



- Для HMM и MEMM вероятности исходящих дуг нормируются для каждого состояния по-отдельности: label bias problem

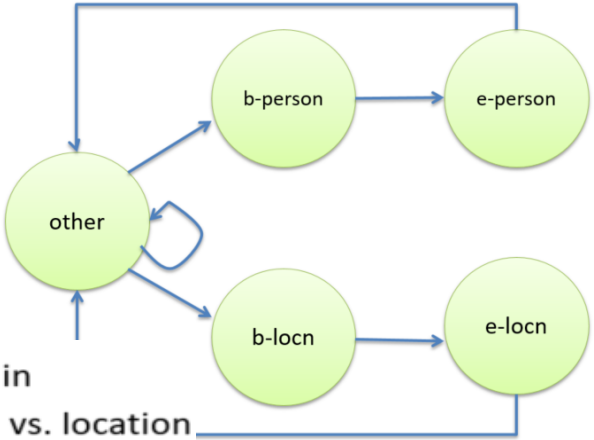
corpus:
Harvey Ford
(person 9 times, location 1 time)
Harvey Park
(location 9 times, person 1 time)
Myrtle Ford
(person 9 times, location 1 time)
Myrtle Park
(location 9 times, person 1 time)

second token a good indicator of person vs. location



Conditional probabilities:

p(b-person | other, w = Harvey) = 0.5
p(b-locn | other, w = Harvey) = 0.5
p(b-person | other, w = Myrtle) = 0.5
p(b-locn | other, w = Myrtle) = 0.5
p(e-person | b-person, w = Ford) = 1
p(e-person | b-person, w = Park) = 1
p(e-locn | b-locn, w = Ford) = 1
p(e-locn | b-locn, w = Park) = 1



Role of second token in distinguishing person vs. location completely lost

Conditional random field

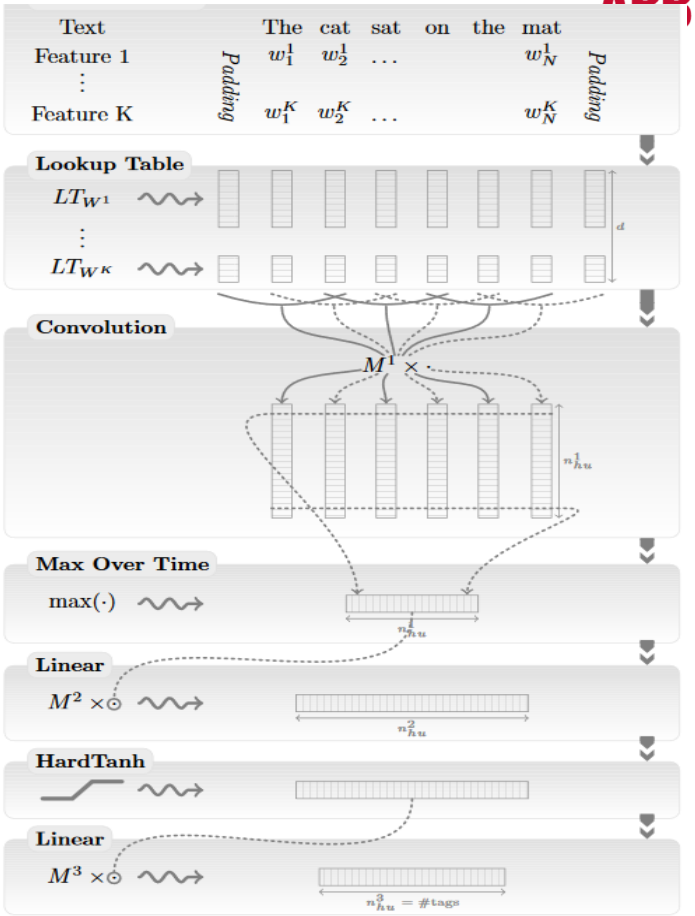
- Входная посл. : $\mathbf{z} = \{\mathbf{z}_1, \cdots, \mathbf{z}_n\}$, выходная посл. для z: $\boldsymbol{y} = \{y_1, \cdots, y_n\}$
- CRF оптимизирует условную вероятность всей выходной посл. При данной входной посл., где $\mathcal{Y}(\mathbf{z})$ множество всех возм. меток, а $\psi_i(y', y, \mathbf{z}) = \exp(\mathbf{W}^T_{y',y}\mathbf{z}_i + \mathbf{b}_{y',y})$

$$p(\boldsymbol{y}|\mathbf{z}; \mathbf{W}, \mathbf{b}) = \frac{\prod_{i=1}^{n} \psi_i(y_{i-1}, y_i, \mathbf{z})}{\sum_{y' \in \mathcal{Y}(\mathbf{z})} \prod_{i=1}^{n} \psi_i(y'_{i-1}, y'_i, \mathbf{z})}$$

- При обучение на множестве $\{(\mathbf{z}_i, \boldsymbol{y}_i)\}$ оптимизируем $\{(\mathbf{z}_i, \boldsymbol{y}_i)\}$ по ММП $L(\mathbf{W}, \mathbf{b}) = \sum_i \log p(\boldsymbol{y}|\mathbf{z}; \mathbf{W}, \mathbf{b})$
- Декодирование на тестовом множестве – нахождение посл. $\boldsymbol{y}^*$ с максимальной условной вероятностью $\boldsymbol{y}^* = \underset{y \in \mathcal{Y}(\mathbf{z})}{\operatorname{argmax}}\, p(\boldsymbol{y}|\mathbf{z}; \mathbf{W}, \mathbf{b})$
- Для linear chain CRF обучение и декодирование эффективно разрешается с помощью алгоритма Витерби (динамика по путям)

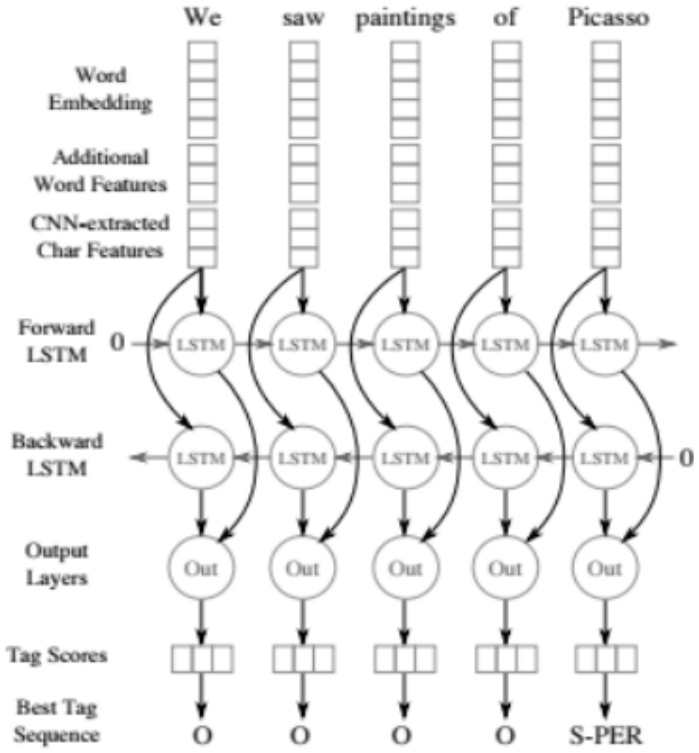
NER: преданья старины глубокой

- Первая удачная попытка решить задачу NER с помощью нейросетей в *Collobert el al (2011), Natural Language Processing (Almost) from Scratch*
- В статье рассмотрены 2 метода – window based и sentence based. Второй лучше, но несколько сложнее
- Признаки – эмбеддинги + ручные признаки (капитализация, POS-таги и т д)
- Модель с газетирами показала SOTA-результат на CoNLL 2003



Шаг в современность – Char CNN + BLSTM + CRF

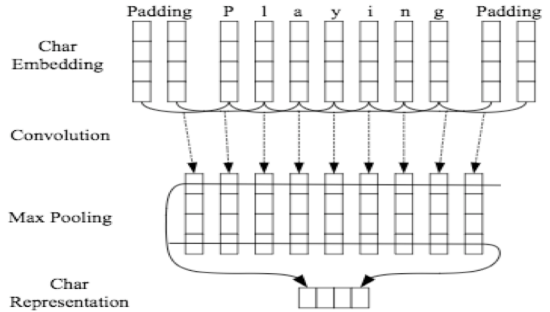
- В данный момент самая популярная архитектура такая:
- Признаки всех токенов предложения (быть может окруженные паддингами и/или контекстом из соседних предложений) подаются в Bidirectional RNN
- В качестве последнего слоя хорошо использовать CRF – он выстраивает метки токенов в согласованные цепочки (и дает прирост f-меры на CoNLL03 ~ 1%)



BLSTM structure for tagging named entities.

Char CNN + BLSTM + CRF: признаки токена

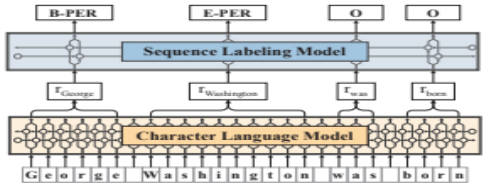
- Признаки токена, как правило, состоят из 3 частей:
  - Словоформенные эмбеддинги. В литературе обычно дообучаются предобученные на большом корпусе эмбеддинги. На практике дообучение большого профита не дает
  - Символьные признаки: эмбеддинги символов каждого токена подаются в CNN (или RNN) небольшого размера. Результат применения CNN или RNN конкатенируется с остальными признаками токена.
  - Могут использоваться дополнительные признаки токена – POS-таги и т. п.



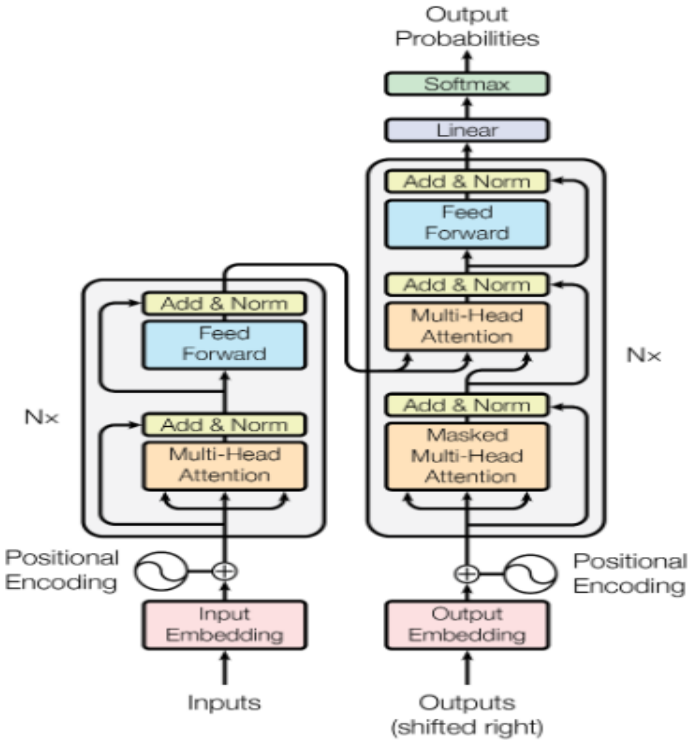
- Применение всей архитектуры в законченном виде появилось в статьях: *Ma and Hovy (2016) End-to-end Sequence Labeling via Bi-direcitonal LSTM-CNNs-CRF* и *Lample el al (2016) Neural Architectures for Named Entity Recognition*

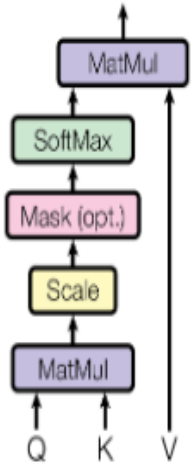
Текущее SOTA

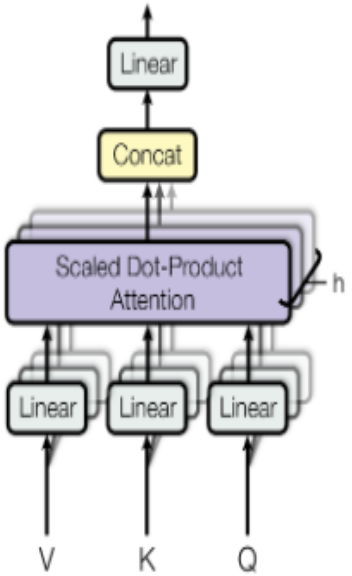
- Простая идея по улучшению качества модели – добавить ELMo к словоформенным эмбеддингам в CharCNN-BLSTM-CRF. Появилось в *Deep contextualized word representations Peters et al., 2018*.
  Качество на CoNLL 2003 – 0.922
- Google сделал языковую модель похожую на ELMo, но на основе трансформера. Появилось в *Devlin et al., 2018 BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding.*
  Качество на CoNLL 2003 – 0.928
- Текущее SOTA Flair embeddings – упрощение языковой модели из ELMo
  Появилось в *Akbik et al., 2018 Contextual String Embeddings for Sequence Labeling.*
  Качество на CoNLL 2003 – 0.931



Трансформер

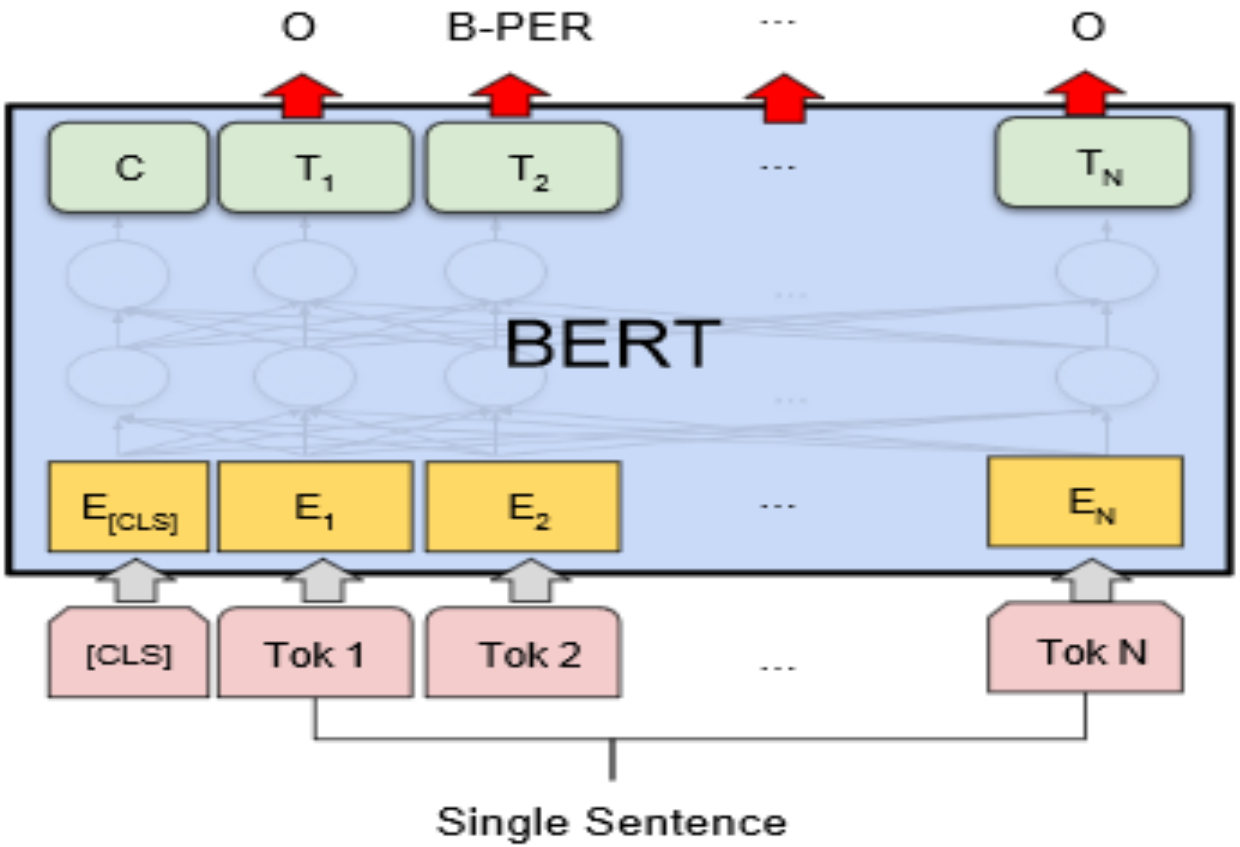






- *Vaswani et al (2017) Attention is all you need*

BERT для NER


O
B-PER
...
O
C
T_1
T_2
...
T_N
BERT
$E_{[CLS]}$
E_1
E_2
...
E_N
[CLS]
Tok 1
Tok 2
...
Tok N
Single Sentence