KU Leuven - NLP 2021-2022
30 November 2021

Exercise Session 4
SRL with CRF, Coreference with ILP

For questions contact:
Toledo Forum for Questions or
nlp@ls.kuleuven.be

1 Semantic Role Labeling with CRF

In this exercise, we work on a sub-task of semantic role labeling: argument classification: Given a certain sentence and the position of the predicate in that sentence, our goal is to build a linear-chain conditional random field (CRF) model that assigns the correct semantic role label to each argument of the predicate. Some toy SRL data could be:

He_{A0} $\text{loves}_{\text{predicate}}$ dogs_{A1} $\text{tremendously}_{\text{AM-MNR}}$

He_{A0} $\text{ran}_{\text{predicate}}$ $\text{fast}_{\text{AM-MNR}}$

The CRF model $P(L|s,j)$ gives the probability for a labeling L for sentence s with predicate at position j, using K different learned weights $\lambda_k \in \mathbb{R}$ and feature functions f_k for $k \in \{1,\ldots,K\}$ where $f_k(\ldots) \in \{0,1\}$ by first calculating a score:

$$score(L|s,j) = \sum_{i=1}^{C(s,j)} \sum_{k=1}^{K} \lambda_k f_k(s,i,j,l_i,l_{i-1})$$

$C(s,j)$ is the number of arguments to be labeled for a given predicate (e.g., for 'loves' in the first sentence, there are 3 arguments to be labeled: 'He', 'dogs' and 'tremendously', so for example

$$C(s = [\text{'He'}, \text{'loves'}, \text{'dogs'}, \text{'tremendously'}], j = 2) = 3$$

l_i is the label for the ith argument in the sentence. For the first sentence, L would be $[\text{A0}, \text{A1}, \text{AM-MNR}]$.

This score is then converted to a probability with a softmax:

$$P(L|s,j) = \frac{exp(score(L|s,j))}{\sum_{L'} exp(score(L'|s,j))}$$

Where L' ranges over all possible answers (including wrong ones).

Question 1 Given the toy data provided above, design three simple feature sets for this task:

- (Argument word)-(role label) feature
- (Predicate word)-(role label) feature
- Role label transition features

How many feature functions can you build for each set, using the two examples as training data? Do you include feature functions that are never observed in the training data? What could be an advantage of doing this? What could be a disadvantage?

Question 2 To train the model, we employ stochastic gradient descent (SGD) on a maximum log likelihood objective: $\mathcal{L}(\lambda) = log(\prod_{s,j} P(L|s,j)) = \sum_{s,j} logP(L|s,j)$
We do this by updating the parameters λ_k for each training example. The derivative w.r.t. a particular weight λ_q is

$$\frac{\partial}{\partial \lambda_q} \log P(L|s,j) = \sum_{i=1}^{C(s,j)} f_q(s,i,j,l_i,l_{i-1}) - \sum_{L'} P(L'|s,j) * \sum_{i=1}^{C(s,j)} f_q(s,i,j,l'_i,l'_{i-1})$$

The weight update for SGD with learning rate α is:

$$\lambda_q^{new} = \lambda_q + \alpha * \frac{\partial}{\partial \lambda_q} \log P(L|s,j)$$

Assuming that only the second example sentence is used for training and only features observed in the training data are used in the model, compute the weight vector λ for 2 iterations when using stochastic gradient descent to optimise the objective function (α = 0.1, and λ is set to zeros as initialisation).

2 Coreference resolution

In this exercise we will try to improve the Berkeley co-reference system[1] by using Integer Linear Programming (ILP). The Berkeley system works as follows: Given N mentions $m_1, ..., m_N$ from a document x, each m_i has an associated random variable a_i taking values in the set of $1, ..., i-1, new$. This variable specifies m_i's selected antecedent or indicates that it begins a new co-reference chain. We call a_i the backpointer of m_i. A setting of all the backpointers, denoted by $\boldsymbol{a} = (a_1, ..., a_n)$, implies an unique set of coreference chains that serve as the system output. An example of such an output would be $(new, 1, new, new, 3, 2)$ for the following sentence, where mentions are indicated by square brackets:

[John]$_{\text{a1=new}}$ plays tennis. [He]$_{\text{a2=1}}$ hits [a ball]$_{\text{a3=new}}$.

[Mary]$_{\text{a4=new}}$ returns [it]$_{\text{a5=3}}$ to [him]$_{\text{a6=2}}$.

To predict the full chain of backpointers a log-linear model of the following conditional distribution is used:

$$P(\boldsymbol{a}|x) \propto exp(\sum_{i=1}^{n} \boldsymbol{w}^T \boldsymbol{f}(i, a_i, x))$$

, where $\boldsymbol{f}(i, a_i, x)$ is a feature function that examines the co-reference decision a_i for m_i with document context x. If a_i = new, the features indicate the suitability of the given mention to be anaphoric or not; when $a_i = j$ for some j, the features express aspects of the pairwise linkage, and examine relevant attributes of the anaphor i or the antecedent j. During training, the model is optimized with a parameterized loss function. The inference is simple and efficient: because $logP(\boldsymbol{a}|x)$ decomposes linearly over mentions, $a_i = argmax_{a_i} P(a_i|x)$. A drawback of computing each a_i locally is that the system does not take into account constraints from mentions outside of the (mention, antecedent) pairs. For example, given three mentions m_1, m_2, m_3, if the system predicts that $a_2 = 1$ and $a_3 = 2$ (i.e., that m_2's antecedent is m_1 and m_3's antecedent is m_2), then m_3 will be automatically inferred as coreferent with m_1. But if there is a clear clue that m_1 and m_3 are not coreferent, leveraging this clue could help avoid the error of linking m_3 to m_2.

Question 3 Design an ILP formulation for computing backpointers globally using $\mathbf{U}$ which is the set of binary indicator variables corresponding to the values assigned to the backpointers. Specifically, $u_{ij} = 1$ iff $a_i = j$ and $u_{ii} = 1$ iff $a_i = NEW$. The objective function should represent the total scores of backpointer assignments over all the mentions.

Question 4 Introduce $\mathbf{V}$, a set of binary variables indicating if two mentions are in the same co-reference chain. For each pair of $j < i$, a variable v_{ij} is added to the ILP model, where $v_{ij} = 1$ iff m_i and m_j are in the same chain. A certain set of backpointer assignments $u_{i,j}$ entails which mentions are in the same chain (encoded by $v_{i,j}$).
In this exercise, we will provide the constraints that define $v_{i,j}$ in terms of $u_{i,j}$. Your job is to turn these constraints into ILP constraints. For example, the constraint of "if a mention starts a new co-reference chain, it is not in the same chain as any of the previous mentions" should be included. In other words, we have: if $u_{ii} = 1$, then $\forall j < i, v_{ij} = 0$. We can represent this as the following ILP constraint: $\forall j < i : u_{ii} + v_{ij} \leq 1$.

Similarly to the example, translate the following constraints to ILP constraints (note that you can need multiple ILP constraints for one constraint):

[1] http://nlp.cs.berkeley.edu/projects/coref.shtml, paper links there are dead, you can find them here: https://aclanthology.org/D13-1203.pdf and https://aclanthology.org/P13-1012.pdf

- If the same backpointer of m_i refers to m_j then the two mentions are in the same chain.
- If $u_{ij} = 1$, then $\forall i > j > k, v_{jk} = v_{ik}$
- If $u_{ij} = 1$, then $\forall i > k > j, v_{ik} = v_{kj}$

Question 5 Each linear inequality that defines the feasible region prohibits certain assignments to the structure. We refer to such constraints as hard constraints. In contrast, a soft constraint merely imposes a penalty on certain assignments rather than prohibiting them. Modify the ILP formulation so that it can work with soft constraints.

Use of V Now that you understand soft constraints, we can explain the use of the $v_{i,j}$ introduced above: The framework of $\mathbf{V}$ variables allows extra coreference constraints to be easily incorporated. For example, consider a soft constraint that is satisfied if there is "an exact string match between a mention and its antecedent". If we want to encourage such matches, for each pair $j < i$ where the two nominal mentions m_i and m_j have an exact string match, we can introduce a constraint indicator variable $c_{exact,i,j}$ and add the (hard) constraint $v_{i,j} + c_{exact,i,j} = 1$ to the ILP model. The result would be that when the exact match constraint is violated (aka there is a pair of mentions with an exact string match, but deemed not in the same chain so for which $v_{i,j} = 0$), ILP would force the corresponding $c_{exact,i,j} = 1$ and how well that proposed solution would satisfy the objective function would be reduced by ρ_{exact}.

$$f_1(s, i, j, l_i, l_{i-1}) = \begin{cases} 1 & \text{if } s[i]=w \ \& \ l_i = y \\ 0 & \text{otherwise} \end{cases}$$

(he, A_0), $(he, A1)$, $(dogn, A1)$,

$$L(\lambda) = \log\left(\prod_{i,j} P(l|s,j)\right) = \sum_{i,j} \log P(l|s,j)$$

(i) $$f_1(s, i, j, l_i, l_{i-1}) = \begin{cases} 1 & \text{if } s[i]=w, \ l_i = y \\ 0 & \text{oth} \end{cases}$$

$(w, y) = \{(He, A0), (fast, AM\text{-}MNR)\}$

(ii) $$f_2 = \begin{cases} 1 & \text{if } s[j]=p, \ l_i = y \\ 0 & \end{cases}$$

$(p, y) = \{(ran, A0), (ran, AM\text{-}MNR)\}$

(iii) $$f_3 = \begin{cases} 1 & \text{if } l_i = l_2 \ \& \ l_{i-1} = l_1 \\ 0 & \end{cases}$$

$(l_1, l_2) = \{(Null, A0), (A0, AM\text{-}MNR)\}$

$L = \{(A0, AM\text{-}MNR), (AM\text{-}MNR, A0)\}$

$F =$

$(f_{He,A0}, f_{fast,AM\text{-}MNR}, f_{ran,A0}, f_{ran,AM\text{-}MNR}, f_{Null,A0}, f_{A0,AM\text{-}MNR})$

$A0 \rightarrow F_{A0} = (1,0,1,0,1,0)$ | $F_{AM} = (0,0,0,1,0,0)$

$F_{AM} = (0,1,0,1,0,1)$ | $F_{A0} = (0,0,1,0,0,0)$

$x_1 = Score([A0, AM\text{-}MNR]|1) = 0$

$x_2 = Score([AM\text{-}MNR, A0]|1) = 0$

$$\frac{2}{\partial \lambda_2} \log P(l|1,j) = \left(1,1,1,1,1,1\right) - \frac{1}{2}\left\{(1,1,1,1,1,1) + (0,0,1,1,0,0)\right\}$$

$$= (0.5, 0.5, 0, 0, 0.5, 0.5)$$

$\lambda_1 = (0.05, 0.05, 0, 0, 0.05, 0.05)$

$$\frac{\partial}{\partial \lambda_2} \log P(l|1,j) = f(1,1,2,A0,Null) - f(1,3,2,AM\text{-}MNR,Null)$$

$$+ P([AM\text{-}MNR, A0]) \times \{f(1,1,2,AM\text{-}MNR,Null) + f(1,3,2,A0,Null)\} +$$

$$P([A0, AM\text{-}MNR]|1) \times \{f(1,1,2,A0,Null) + f(1,3,2,A0,Null)\}$$

$$= (1,1,1,1,1,1) - 0.45 \times (0,0,1,1,0,0) + 0.55 \times (1,1,1,1,1,1)$$

$$= (0.45, 0.45, 0, 0, 0.45, 0.45)$$

$$\lambda_2 = (0.095, 0.095, 0, 0, 0.095, 0.095)$$