COL776 - Assignment 1 (Part B)

Akshay Kumar Gupta
2013CS50275

Implementation Language: Python

Q1a. Summary of the paper "Maximum Entropy Markov Models for Information Extraction and Segmentation" by McCallum, Freitag and Pereira.

Motivation

Hidden Markov Models (HMMs) suffer from two main problems:

- Many tasks like text segmentation and entity recognition can be improved by having a richer representation of observations than is allowed by HMMs and it would be beneficial to allow observations to be represented by arbitrary overlapping features (like word, capitalisation, POS, formatting etc. in the case of segmentation).
- HMMs uses a joint model and its parameters are set to maximise the likelihood of the observation sequence. However, in most text-based tasks, we need to predict the state sequence given the observation sequence.

Model

Maximum Entropy Markov Models (MEMMs), like HMMs, also have hidden states and observed variables. However, the transition and emission parameters of HMMs are replaced by a single function $P(s|s', o)$, which is the probability of the current state s given the previous state s' and the current observation o. Thus MEMM is a discrimative model as opposed to HMM which is a generative model. Further, $P(s|s', o)$ is split into $|S|$ separately trained functions $P_{s'}(s|o)$ (one for each s'), and each of these functions follow an exponential model. This exponential model is defined by a set of binary features over the current state s and the current observation o.

$$P_{s'}(s, o) = \frac{1}{Z(o,s')} \exp\left(\sum_a \lambda_a f_a(o, s) \right)$$

Here, λ_a is the parameter to be learnt for each feature a, and $Z(o, s')$ is the normalising factor.

This exponential distribution satisfies the constraint that the expected value of each feature in the learned distribution is the same as its average on the training observation sequence $o_1 \ldots o_m$.

Advantages of MEMMs

1. They model the conditional distribution of hidden states given observed variables, which can make inference and learning more tractable.
2. They allow overlapping of features in the representation of observations, which allows for a richer set of features to be modelled in comparison to HMMs.

Q1b. Summary of the paper "Conditional Random Fields: Probabilistic Models for Segmenting and Labeling Sequence Data" by Lafferty, McCallum and Pereira.

Motivation

MEMMs and some other discriminative models suffer from a problem known as the 'label bias problem'. In MEMMs, each source state has a separate normalised probability distribution of next state given observation. Because of this, all the mass arriving at a state is distributed among the possible successor states. The observations can only influence which states get how much mass, not how much mass gets passed on. This leads to a bias towards states with fewer outgoing transitions. If a state has only one successor, then the observation at that point is effectively ignored. CRFs aim to address this issue while retaining all the benefits of MEMMs over HMMs.

Terminology

- X is a random variable over data sequences to be labeled

- Y is a random variable over label sequences. Components Y_i of Y range over a finite label alphabet $\mathcal{Y}$.

Model

Let $G = (V, E)$ be a graph such that $Y = (Y_v)_{v \in V}$, so that Y is indexed by the vertices of G. Then (X, Y) is a conditional random field in case, when conditioned on X, the random variables Y_v obey the Markov property with respect to the graph: $P(Y_v|X, Y_w, w \neq v) = p(Y_v|X, Y_w, w \sim v)$,where $v \sim w$ means that w and v are neighbours in G.

The paper assumes the graph G over Y to be a tree. In this case, the cliques in G are only the edges and vertices of G. Hence, for the conditional random field (Y, X), the distribution over the label sequences Y given X has the form:

$$P_\theta(y|x) \propto \exp\left(\sum_{e \in E,k} \lambda_k f_k(e, y|_e, x) + \sum_{v \in V,k} \mu_k g_k(v, y|_v, x)\right)$$

or
$$P_\theta(y|x) = \tfrac{1}{Z_x} \exp\left(\sum_{e \in E,k} \lambda_k f_k(e, y|_e, x) + \sum_{v \in V,k} \mu_k g_k(v, y|_v, x)\right)$$

where x is a data sequence, y is a label sequence, $y|_S$ is the set of components of y associated with the vertices in the subgraph S, f_k and g_k are a given set of features defined either over variables in Y or variables in both X and Y, and Z_x is the normalisation factor that depends only on the observed sequence x.

The paper further assumes that the dependencies of Y, conditioned on X, form a chain. Then for each position i in the observation sequence x, we can define the $|\mathcal{Y}| \times |\mathcal{Y}|$ matrix $M_i(x) = [M_i(y', y|x)]$ by

$$\begin{aligned}
M_i(y', y|x) &= \exp(\Lambda_i(y', y|x)) \\
\Lambda_i(y', y|x) &= \sum_k \lambda_k f_k(e_i, Y|_{e_i} = (y', y), x) + \sum_{v \in V,k} \mu_k g_k(v, Y|_{v_i} = y, x)
\end{aligned}$$

Then the normalisation function is

$$Z_\theta(x) = \left(\prod_{i=1}^{n+1} M_i(x)\right)_{(0,n+1)}$$

Hence the conditional probability of a label sequence y given an observation sequence x is

$$P_\theta(y|x) = \frac{\left(\prod_{i=1}^{n+1} M_i(y_{i-1}, y_i|x)\right)}{\left(\prod_{i=1}^{n+1} M_i(x)\right)_{(0,n+1)}}$$

Advantages of CRFs

1. The class of CRFs is much more expressive than HMMs, because it allows arbitrary dependencies between the observed variables.
2. They model the conditional distribution of hidden states given observed variables, which can make inference and learning more tractable.
3. The features do not need to completely specify a state or observation, so it is expected that the model can be estimated from less training data.
4. The loss function is convex.
5. The probability distribution of state transition is normalised across all possible source states, so the label bias problem does not occur.

Experimental Results

1. It was confirmed experimentally using a synthetically generated dataset that CRFs do not face the label bias problem while MEMMs do.
2. In a POS tagging experiment it was observed that CRF outperform MEMMs, and adding overlapping features for the observations (which cannot be done in an HMM) allowed both CRF and MEMM to outperform HMM significantly.

Q2a. Data Structures used:

- A dictionary (hash table) D is used to map each character to a distinct number in the range 0-9 (as there are only 10 characters in the vocabulary).
- The transitional probabilities are represented by a two-dimensional 10x10 vector T of floats where $T[i][j]$ is the probability $P(y_{t+1} = b|y_t = a)$, where $D[a] = i$ and $D[b] = j$.
- The emission probabilities are represented by a two-dimensional 1000x10 vector E of floats where $E[i][j]$ is the probability $P(x_t = i|y_t = a)$, where i is the image number and $D[a] = j$.

Q2b. Accuracy table:

- OCR Model (OCR Factors)

Dataset	Character Accuracy	Word Accuracy	Average Log-Likelihood
small/images.dat	53.92 %	8.65 %	-7.808
large/images1.dat	58.39 %	11.11 %	-7.876
large/images2.dat	57.25 %	10.00 %	-7.874
large/images3.dat	57.24 %	9.91 %	-7.865
large/images4.dat	57.57 %	11.47 %	-7.869
large/images5.dat	58.53 %	11.56 %	-7.857

- Transition Model (OCR and Transition Factors)

Dataset	Character Accuracy	Word Accuracy	Average Log-Likelihood
small/images.dat	66.27 %	25.96 %	-7.097
large/images1.dat	68.04 %	24.04 %	-7.175
large/images2.dat	67.68 %	24.17 %	-7.174
large/images3.dat	67.87 %	24.68 %	-7.167
large/images4.dat	68.24 %	24.68 %	-7.170
large/images5.dat	68.45 %	26.69 %	-7.158

- Combined Model (OCR, Transition and Skip Factors)

Dataset	Character Accuracy	Word Accuracy	Average Log-Likelihood
small/images.dat	71.17 %	35.57 %	-6.279
large/images1.dat	70.83 %	31.48 %	-6.271
large/images2.dat	70.72 %	31.80 %	-6.271
large/images3.dat	70.62 %	31.94 %	-6.265
large/images4.dat	70.77 %	31.85 %	-6.267
large/images5.dat	71.06 %	33.31 %	-6.257

- Examples of words in the small dataset that were incorrectly predicted by OCR Model and corrected by Transition Model : ado, hent, reader, renter, toston, tote
- Examples of words in the small dataset that were incorrectly predicted by OCR Model, partially corrected by Transition Model and fully corrected by Combined Model : noon, ratoon, seeder

Extra Credit : I tried squaring the Transition and OCR parameters, varying the skip factor, and combinations of the above, but the accuracy decreased in all of these cases (on the small dataset). Scaling all the transition/OCR factors will result in the same distribution so there is no point in doing that.

Discussed assignment with : Barun Patra, Haroun Habeeb, Kabir Chhabra