Text Analytics

Parts of Speech - Hidden Markov Model

Ramaseshan Ramachandran

1. Parts of Speech Hidden Markov Algorithm

Parts of Speech
Initialization

HIDDEN MARKOV MODEL

The POS-HMM can be used to compute the probability of a given sequence of words, as well as the most likely sequence of POS tags for a given sentence.

Components

- **States** - The set of possible speech tags
- **Observations** - The words in the vocabulary
- **Transition Probability** - The probability of transitioning from one state to another
- **Emission Probability**- The probability of observing a particular word given a particular state

TRANSITION EXAMPLES

Transitioning between common POS tags

- $\mathrm{P}(\mathrm{Noun} \mid \mathrm{Determiner})$: This represents the probability of a noun following a determiner (e.g., "the", "a"). In English, this is a very high probability, as determiners typically introduce nouns.
- $\mathrm{P}(\mathit{Verb} \mid \mathrm{Noun})$: This represents the probability of a verb following a noun. This is also quite common, as verbs often describe actions performed by the noun
- $\mathrm{P}(\mathit{Adjective} \mid \mathrm{Comma})$: This represents the probability of an adjective following a comma. This is common for listing multiple adjectives describing the same noun

EMISSION PROBABILITIES

Common words and their likely POS tags

- P("the" | Determiner): Very high probability because "the" is almost always a determiner
- P("dog" | Noun): High probability
- P("run" | Verb): High probability

Ambiguous words:

- P("book" | Noun): Mostly high probability
- P("book"|Verb) Depending on the context, this could be high
- P("dust"|Noun): High probability (around 0.8-0.9) because "dust" is mostly a noun. P("dust" | Verb): Lower probability (around 0.1-0.2) because "dust" can also be a verb in specific cases.

UNSUPERVISED LEARNING

Can we use unsupervised Learning?

- **Initialization**: Assign initial POS tags randomly or based on some heuristic.
- **Expectation Step:** Estimate the probability of each word belonging to different POS tags based on the current set of POS tags.
- **Maximization Step:** Update the POS tags based on the probabilities obtained in the expectation step.
- **Repeat:** Iterate between the expectation and maximization steps until convergence.

Did it work? 🤔

INITIALIZATION

The model is initialized with the learned parameters obtained during the training phase. Input Sentence: Given a new sentence with words $W = w_i, w_2, w_3, \ldots, w_m$, the goal is to assign the most likely sequence of POS tags $T = t_1, t_2, t_3, \ldots, t_m$ to the corresponding words in W

- Define the set of POS tags: $T = \{t_1, t_2, \ldots, t_m\}$.
- Initialize the state transition probabilities A_{ij}, where A_{ij} represents the probability of transitioning from t_i to t_j.
- Initialize the emission probabilities B_{jk}, where B_{jk} represents the probability of observing word k given the POS t_j.
- Initialize the initial state probabilities π_i, where π_i represents the probability of starting with POS t_i.

MARKOV CHAIN

Definition

A Markov chain is a stochastic process characterized by the Markov property: the probability of transitioning to the next state depends only on the current state, not on the history of previous states.

TRAINING AND TAGGING

Training

- Train the HMM on a labeled dataset of sentences and their corresponding POS tags.
- Estimate the transition probabilities A_{ij}, emission probabilities B_{jk}, and initial state probabilities π_i using the training data.

POS Tagging

- Given a new sentence with words $W = \{w_1, w_2, \ldots, w_m\}$:
- Initialize the Viterbi matrix V with dimensions $n \times m$, where n is the number of POS tags and m is the number of words in the sentence.
- Fill in the Viterbi matrix using the forward algorithm:

$$V_{ij} = \max_k (\pi_k \times B_{kj} \times A_{ki}) \times \text{score of previous state}$$

- Backtrack to find the optimal sequence of POS tags

VITTERBI ALGORITHM

The Viterbi algorithm is a dynamic programming algorithm used for finding the most likely sequence of hidden states (or labels) in a Hidden Markov Model (HMM) given an observed sequence of data. In the context of Parts of Speech (POS) tagging, the hidden states are the POS tags, and the observed sequence is the sequence of words in a sentence.