

# Punctuation Prediction using Conditional Random Fields

Suvir Jain,  
Computer Science  
Department  
University of California, San  
Diego  
suj011@cs.ucsd.edu

Gaurav Saxena  
Computer Science  
Department  
University of California, San  
Diego  
gsaxena@cs.ucsd.edu

Kashyap Tumkur  
Computer Science  
Department  
University of California, San  
Diego  
gsaxena@cs.ucsd.edu

## ABSTRACT

## 1. INTRODUCTION

## 2. PRELIMINARIES

### 2.1 Assumptions

Tag: SPACE to show No punctuation Tag START to mark the beginning of the tag sequence Tag: PERIOD, QUESTION MARK to mark the end of the tag sequence

## 3. IMPLEMENTATION

### 3.1 Algorithms for CRF

*Viterbi path algorithm for Inference*  
*Proof of correctness*

*Forward and Backward Vectors*

*Proof of correctness.* Find  $Z$  using both  $\alpha$  and  $\beta$  vectors

#### 3.1.1 Learning using SGD

*Proof of correctness.* Prove derivatives are correct

### 3.2 Learning using Collins Perceptron

*Proof of correctness*

#### 3.2.1 Feature Functions

1. -ing words
2. interrogative words, subject-verb inversion
3. connectives like however
4. conjunctions
5. Interjections

### 3.3 Data cleaning

### 3.4 Overfitting

#### 3.4.1 Validation

#### 3.4.2 Regularization?

#### 3.4.3 Feature Scaling?

#### 3.4.4 Randomization

Can we have our feature functions output only values between 0 and 1

## 4. EXPERIMENTS

### 4.1 Experiment 1: Learning using SGD

Charts proving convergence of SGD Charts showing accuracy on training, validation and test set

### 4.2 Experiment 2: Learning using Collins Perceptron

Charts proving convergence of CP Charts showing accuracy on training, validation and test set

## 5. OPTIMIZATIONS

Optimizations done to the algorithm to improve running time like:

1. Run the algorithms on a random smaller subset to speed up implementation
2. Use MATLAB profiler to find bottlenecks and remove them

## 6. CONCLUSION

### 6.1 Lessons Learnt

### 6.2 Discussion

Comparison of two approaches in terms of time to converge, their accuracies on training, validation and test sets

## 7. REFERENCES