

# CS626 Assignments

Abhirut Gupta

Mandar Joshi

Piyush Dungarwal

(Group No. 6)

# Trigram Accuracy

FoldNo	Total Tags	Correct Tags	Accuracy
1	225043	214515	95.322
2	71502	68210	95.395
3	96749	92599	95.710
4	359255	342266	95.271
5	110236	105403	95.616

# Confusion Matrix

- CJS : accuracy = 87.43 %
  - Confused with : PRP = 9.35 %
  - Eg. “As” CJS in “diagnosed as being HIV positive”  
PRP in “give AIDS greater recognition not as a disease”
- VVB : accuracy = 70.72 %
  - Confused with : VVI = 12.97 %
  - Eg. Forget, send, live, return, etc.

# Confusion Matrix

- AV0 : accuracy = 87.27 %
  - Confused with : PRP = 4.67 %
- VVD : accuracy = 88.98 %
  - Confused with : VVN = 6.49 %
  - Eg. Sent, lived, returned, etc.

# Insight

- Nouns and verbs are often confused because,
  - Most verbs can also be used as nouns.
    - E.g. laugh, people, etc.
- Adverbs (which often precede verbs) can also be used as adjectives (which often precede nouns)
  - E.g. fast, slow, etc.

# Next word prediction

Model	Accuracy	Perplexity
Tagged Model	11.32 %	11.83
Untagged Model	4.67 %	22.87

$$\begin{aligned}(W3, T3)^* &= \text{Argmax}_{(W3, T3)} (P(W3, T3 \mid W2, W1, T2, T1)) \\ &= \text{Argmax}_{(W3, T3)} (P(T3 \mid W2, W1, T2, T1) * P(W3 \mid W2, W1, T3, T2, T1)) \\ &= \text{Argmax}_{(W3, T3)} (P(T3 \mid T2, T1) * P(W3 \mid W2, W1, T3))\end{aligned}$$

# Generative vs. Discriminative

## (Bigram)

FOLD	Discriminative Model	Generative Model
1	79.90 %	82.89 %
2	81.08 %	80.79 %
3	81.91 %	81.02 %
4	80.84 %	83.45 %
5	81.80 %	80.95 %
Total	80.85 %	82.49 %

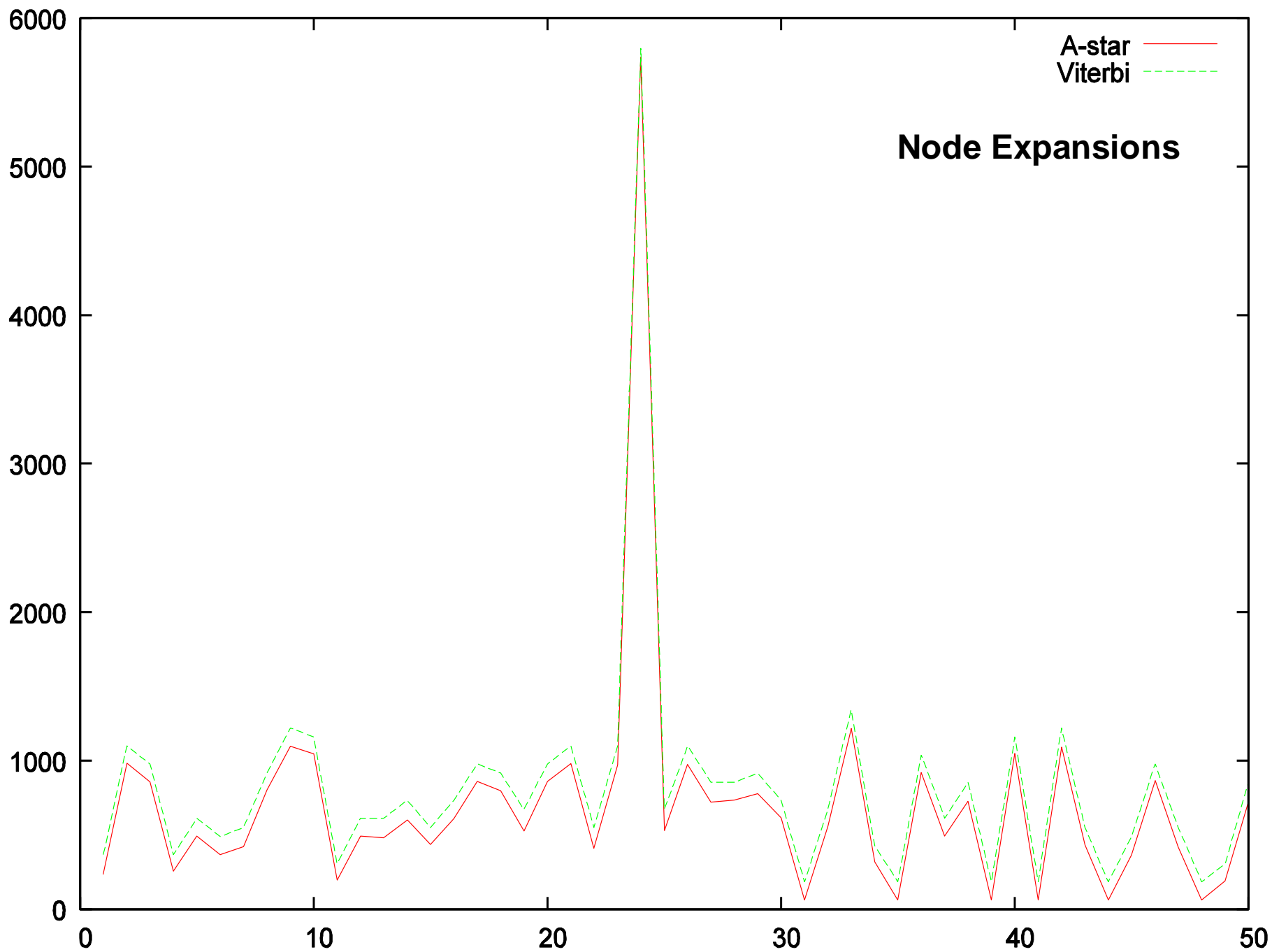
- Generative model possibly gives better accuracy than Discriminative model because, we model Transition and Lexical probability separately in the former model.



# A\* Results

A* accuracy (With unknowns)	A* accuracy	Viterbi
52.3 %	94.06 %	95.39 %

<sup>1</sup> Trigram Viterbi after applying Capitalization



- In Trigram Viterbi implementation, Capitalization is applied outside the algorithm.
- This results in certain words being tagged differently by two algorithms.
- From the graph,  $A^*$  expands fewer nodes than Viterbi. However statistics for node expansion do not take into account the comparisons made in the Viterbi algorithm

# Beam search algorithm

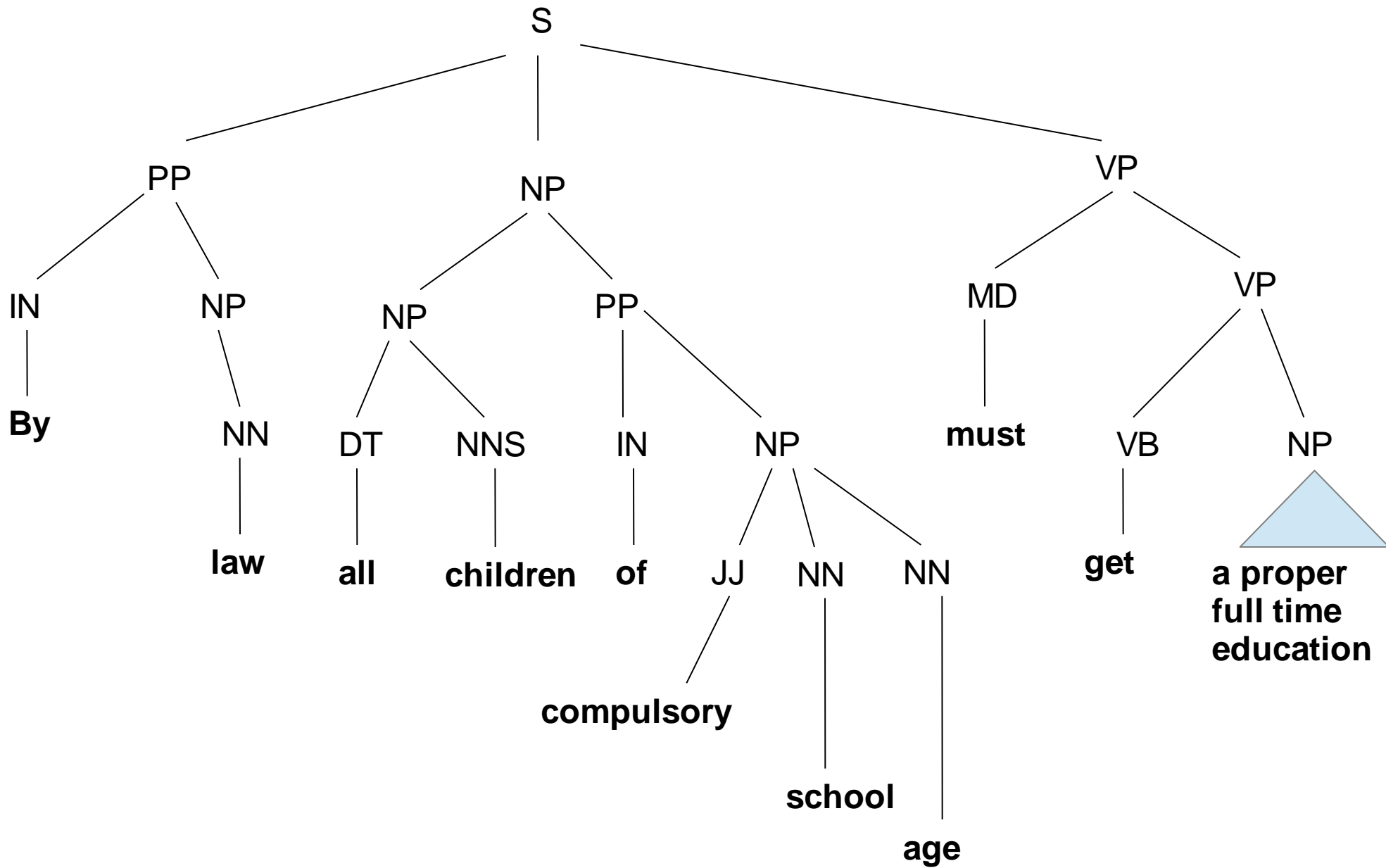
- An upper bound is maintained on the size of Open list of nodes
- Results for size = 50:
  - Reduction in execution time : 56.62%
  - Accuracy : 78.28 %
- Results for size = 25:
  - Reduction in execution time : 76.50%
  - Accuracy : 63.24 %

# Projection of Parse Trees

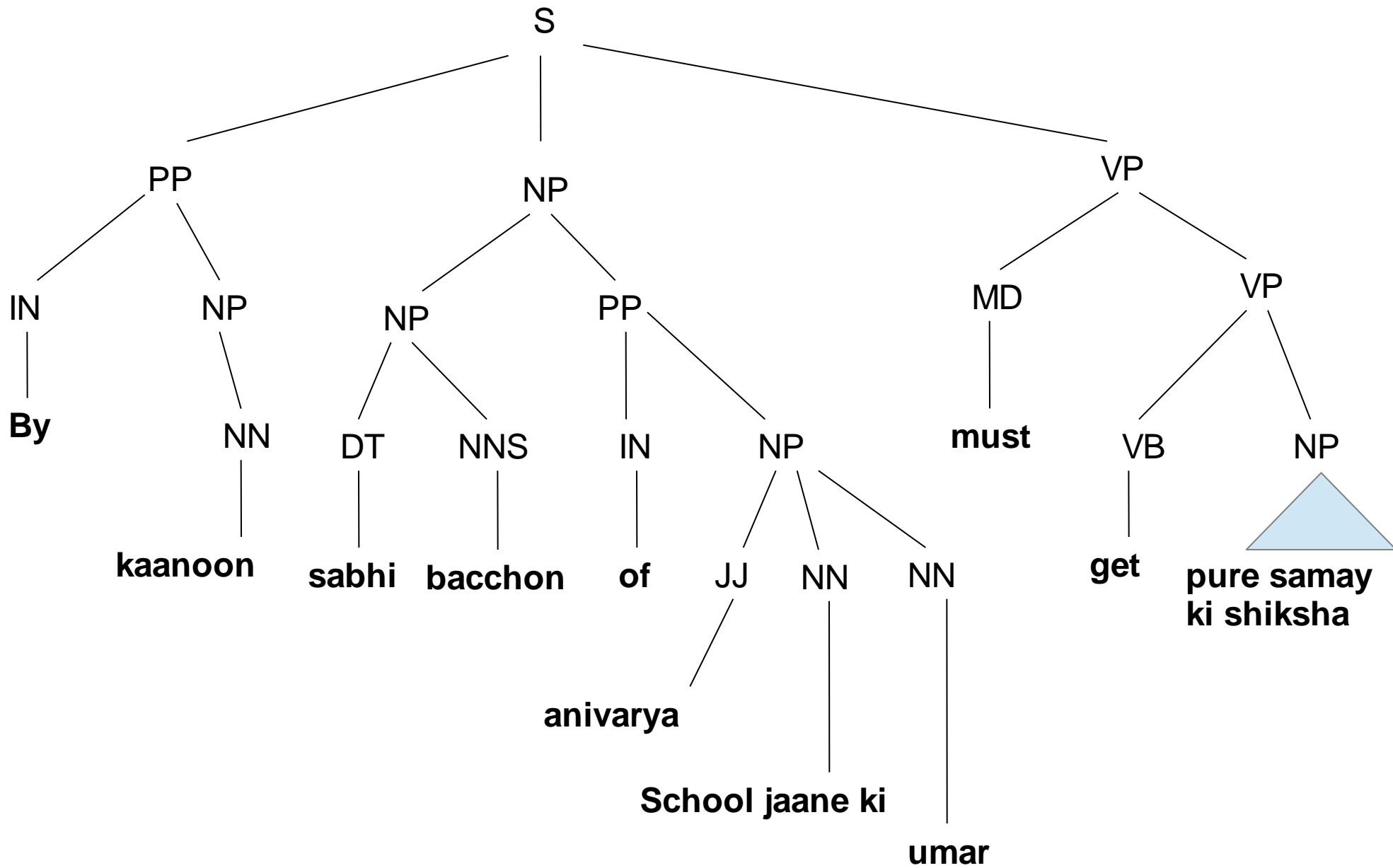
# Our Approach

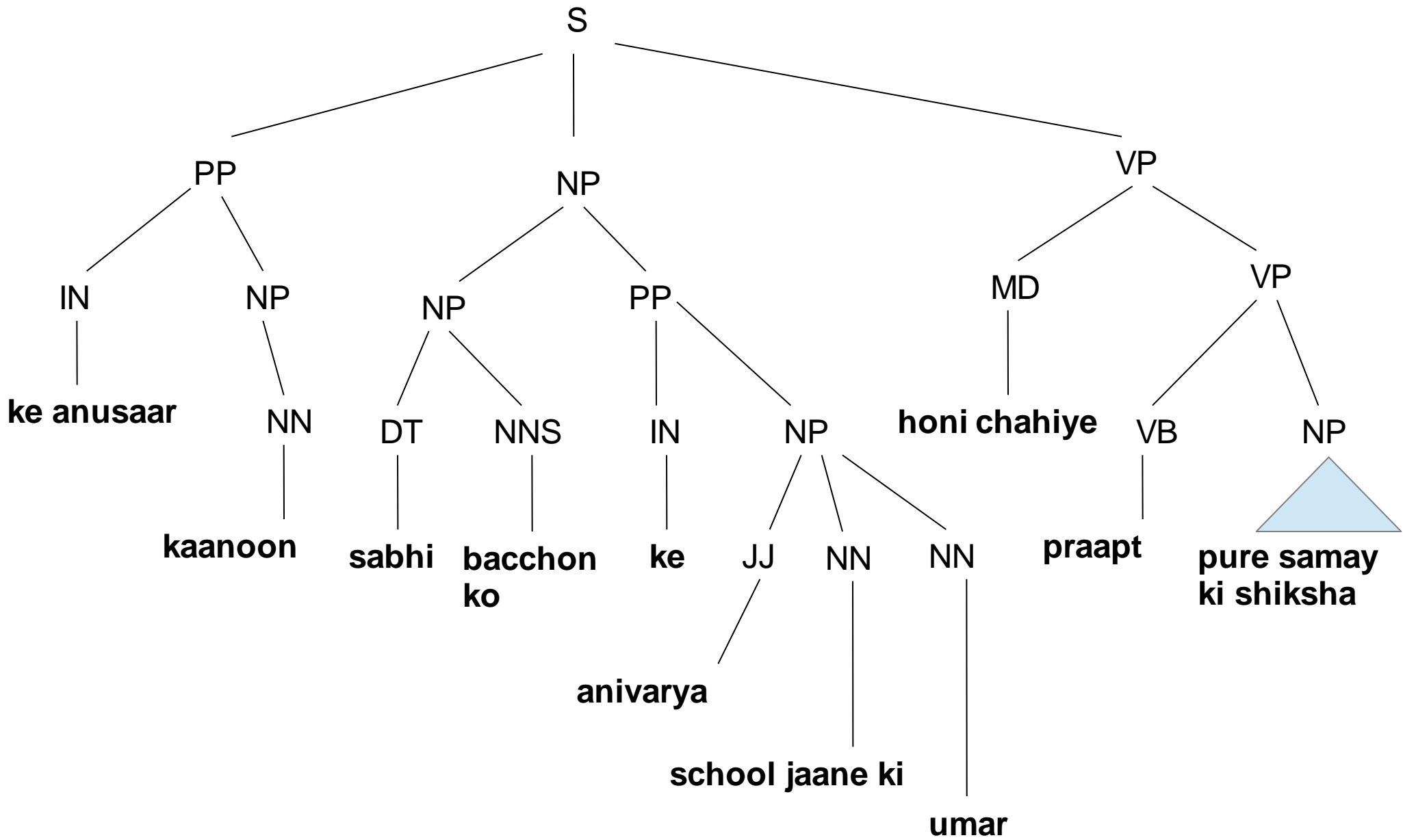
- From the English parse tree, we obtain the Noun Phrases.
- Using an English-Hindi dictionary and the given translation, find corresponding groups in the Hindi sentence, and replace the English groups by the Hindi ones.

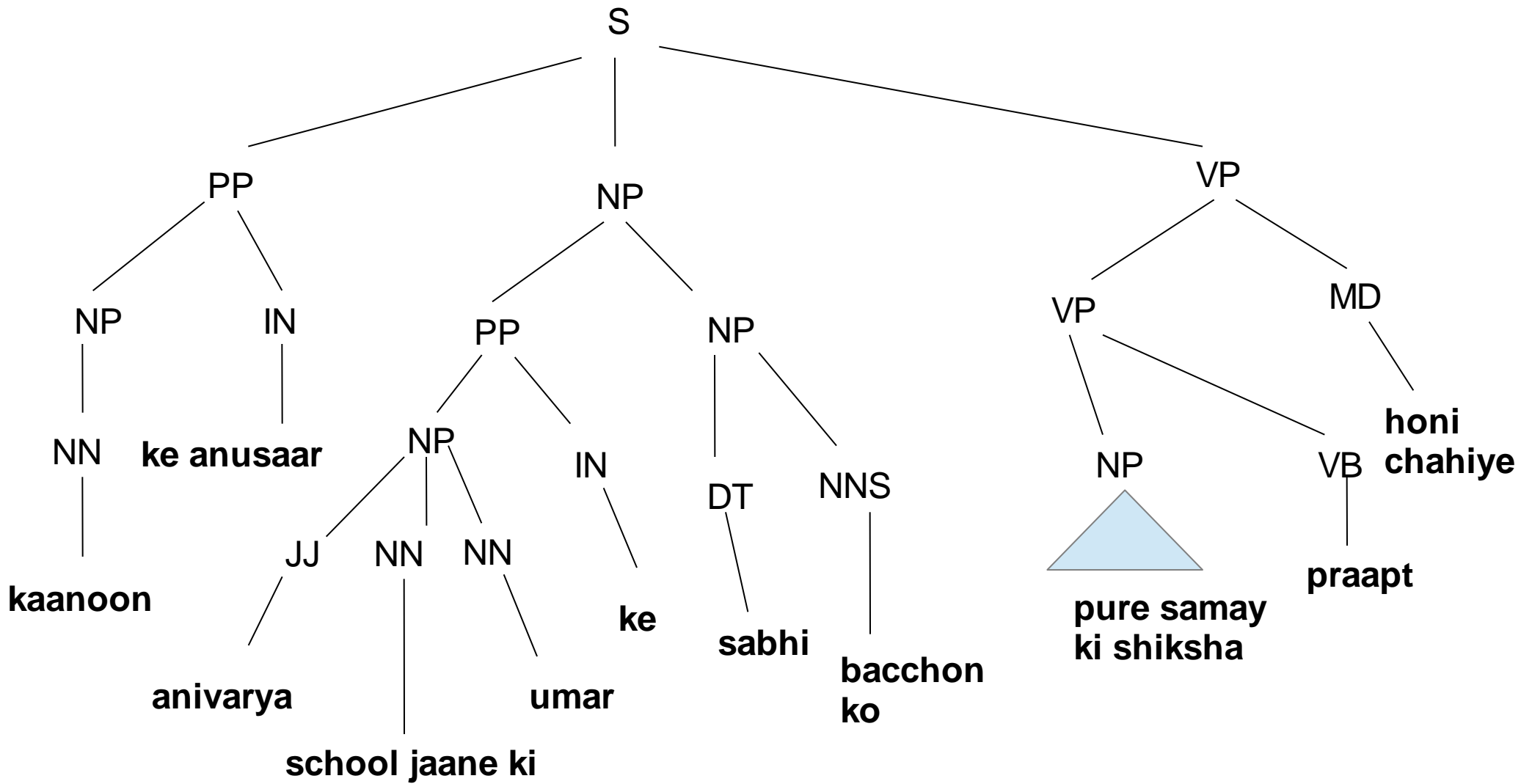
- Replace the remaining words by corresponding Hindi words from the sentence.
- Now, apply hard-coded inversion rules to the tree.
  - Example – PP  $\rightarrow$  P NP in english is inverted as  
PP  $\rightarrow$  NP P in hindi











# Challenges

- Translated sentences may have a different structure altogether.
- It may not be possible to detect similar phrases.

Example -

- आपका बच्चा ड्रग्स या सॉल्वेंट्स लेने के चक्कर में क्यों पड़ गया है, यदि आप उसके कारणों को समझते हैं, तो इस विषय में अपने बच्चों से बात करना आपके लिए काफी आसान होगा ।
- If you understand the reasons why a child can get involved with drugs and solvents, it's much easier for you to talk to your children about it.

# Drawbacks

- Parse trees produced may be “approximate”
- Due to morphological differences it may become difficult to produce tags for extra words created in translation. Example – “ko”, “jaane ki” produced in the previous example.

# Yago

- Interesting relations:
  - Sachin Tendulkar and Rahul Dravid have both won the Arjuna Award.
  - Dev Anand was born in Mumbai, died in London.
  - Rohit Sharma and Vikram Pandit were born in Nagpur.