Final Project

11761 Language and Statistics

MEMBERS

- Abhinav Arora
- Apoorva Bansal
- Shachi Paul
- Vishrav Chaudhary

OUTLINE

- Introduction
- Feature Engineering
  - Bag of words
  - TF-IDF
  - Type-token Ratio
  - Average log-likelihood of n-gram language models
  - POS-tags
  - Topic modeling
- Classification
- Results
- Conclusion
- References

OUTLINE

- Introduction
- Feature Engineering
  - Bag of words
  - TF-IDF
  - Type-token Ratio
  - Average log-likelihood of n-gram language models
  - POS-tags
  - Topic modeling
- Classification
- Results
- Conclusion
- References

INTRODUCTION

GOAL

To investigate, discover and exploit deficiencies in the conventional trigram language model, using statistical methods.

METHOD

Build a machine learning classifier capable of discovering the deficiencies in the conventional trigram language model, and hence, discriminating between fake and real articles.

OUTLINE

- Introduction
- Feature Engineering
  - Bag of words
  - TF-IDF
  - Type-token Ratio
  - Average log-likelihood of n-gram language models
  - POS tags
  - Topic modeling
- Classification
- Results
- Conclusion
- References

FEATURE ENGINEERING

Bag of words:

- Frequency of word-occurrence in article used as feature.
- Did not work well.
- Doesn’t consider distributional differences of sentences.

FEATURE ENGINEERING

TF-IDF:

- Penalizing commonly occurring terms by their inverse document frequency.
- Did not add much information to our model.
- Doesn’t capture contextual meaning of sentences.

FEATURE ENGINEERING

Type-token ratio:

- A type-token curve is a very good measure of language identification.
    - Intuition:
        - Considers distributional differences of sentences between the training and development articles.
- Did not add much information to our classifier
    - Possible reason:
        - Very few sentences in each article in the development set as compared to that of the training set.

FEATURE ENGINEERING

Average log-likelihood (n-gram models):

- Trained 3-gram, 4-gram, 5-gram, 6-gram and 7-gram models on Broadcast News Dataset using CMU Statistical Modelling Toolkit.

- Used the Average Log Likelihood of the articles with these models as the features.

- Results showed that these were the best features.

FEATURE ENGINEERING

POS tags:

- Similar approach - Assignment 7 (Decision Tree Language Models).
- Used part of speech tags of the sentences in the articles as a feature.
- This was a good feature, but wasn’t able to perform as well as the Perplexity of N-gram language models.

FEATURE ENGINEERING

Topic Modeling:

- N-gram models only capture the local relations

- Use topic modeling to exploit the semantic relations between different words spread throughout the article

FEATURE ENGINEERING

Topic Modeling:

- Used LDA to identify 50 different topics
- Extracted features:
  - **Word Percentage per Topic:**
    - Percentage of words in an article that belong to a particular topic
    - Intuition: indicates distribution of topics across an article
  - **Topical Entropy of an Article:**
    - Topical entropy of that article with respect to the 50 latent topics
    - Intuition: real article - few topics - the entropy of its topical distribution should be low

FEATURE ENGINEERING

Topic Modeling:

- Didn’t work!
  - Possible reason:
    - quality of the articles in the development set - too short! (<s> LET'S TAKE </s>)
    - topical distribution assigned to such short articles doesn't really give any information.

OUTLINE

- Introduction
- Feature Engineering
  - Bag of words
  - TF-IDF
  - Type-token Ratio
  - Average log-likelihood of n-gram language models
  - POS-tags
  - Topic modeling
- Classification
- Results
- Conclusion
- References

CLASSIFICATION

Classifier:

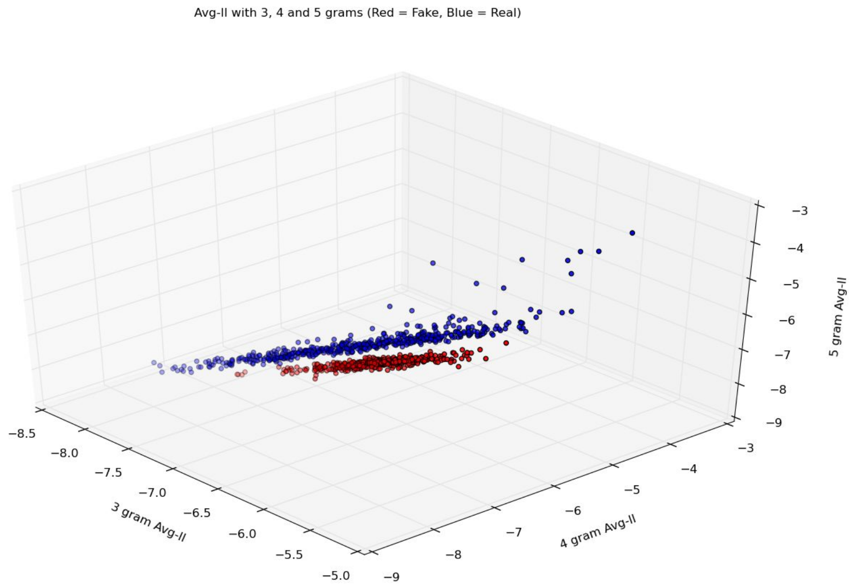
Linear Support Vector Machine(SVM)

Features:

3, 4, 5, 6, 7 grams average log-likelihood

CLASSIFICATION

Data is linearly separable!



CLASSIFICATION

- Graphs were pretty good for 3-grams and 4-grams, but...
- Cross validation on the development set was not giving good accuracy.
- Used up to 7 grams.

OUTLINE

- Introduction
- Feature Engineering
  - Bag of words
  - TF-IDF
  - Type-token Ratio
  - Average log-likelihood of n-gram language models
  - POS-tags
  - Topic modeling
- Classification
- Results
- Conclusion
- References

RESULTS

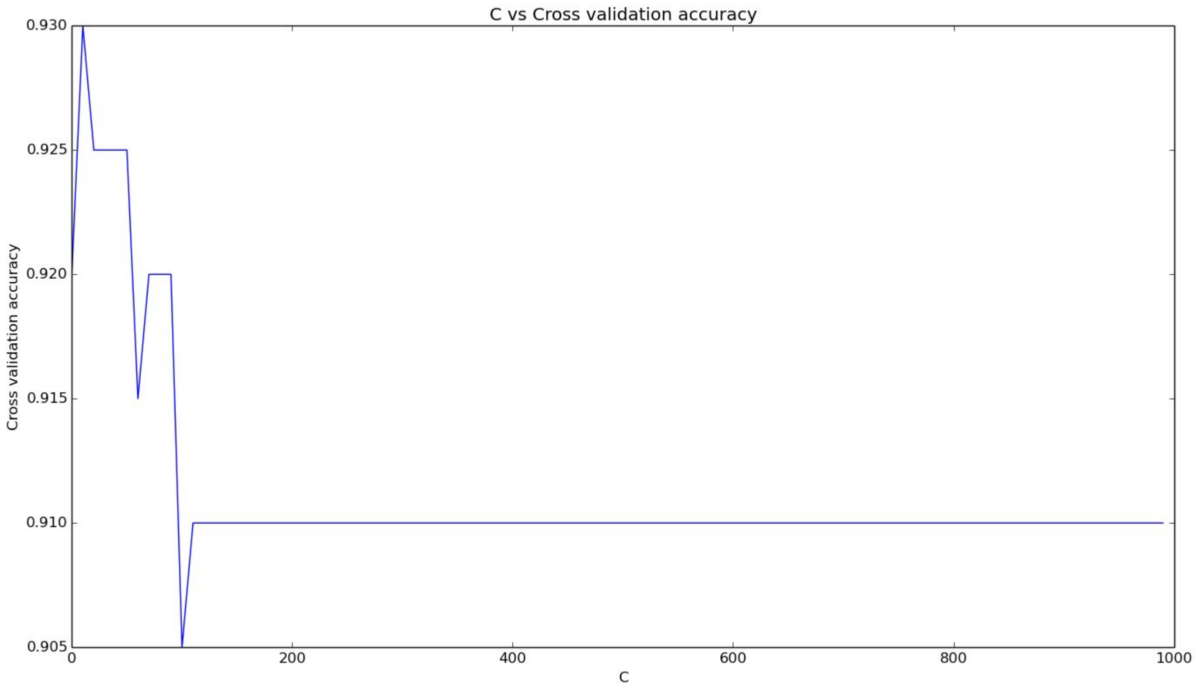
Parameters:

- C = 10
- Kernel = Linear

Cross-validation Accuracy: 99.5%

Accuracy on development set: 93%

RESULTS



C vs Cross validation accuracy
0.930
0.925
0.920
0.915
0.910
0.905
Cross validation accuracy
0
200
400
600
800
1000
C

OUTLINE

- Introduction
- Feature Engineering
  - Bag of words
  - TF-IDF
  - Type-token Ratio
  - Average log-likelihood of n-gram language models
  - POS-tags
  - Topic modeling
- Classification
- Results
- Conclusion
- References

CONCLUSION

- Bag of word features don’t work well.
  - unable to account for the contextual meaning of sentences in the data set
- Topical modelling could not add any new information.
  - Articles in the development set were too short
- Balancing the dataset didn’t work.
  - the points were very close in feature space
- SVM performed better than Decision trees and Random Forest classifiers.
- Final weights - 3-grams and 7-grams are important.

OUTLINE

- Introduction
- Feature Engineering
  - Bag of words
  - TF-IDF
  - Type-token Ratio
  - Average log-likelihood of n-gram language models
  - POS-tags
  - Topic modeling
- Classification
- Results
- Conclusion
- References

References

[1] Vladimir Vapnik and Corinna Cortes. Support vector networks. Machine Learning, 20:273–297, 1995.

[2] Gensim – Latent Dirichlet Allocation (https://radimrehurek.com/gensim/models/ldamodel.html).

[3] H. Misra and F. Yvon. Using lda to detect semantically incoherent documents. In Proceedings of CoNLL, 2008.

[4] Statistical Language Modeling Toolkit (http://svr-www.eng.cam.ac.uk/~prc14/toolkit.html)