# IntelliWeb

## Intelligent web page classification

By:

FNU Vivek

Rithish Koneru

Aayushi Agrawal

Ravindersingh Rajpal

Group: P08

# MOTIVATION

- Explosive growth on the Internet, with millions of web pages on every topic
- Important task is to collect relevant information from Web
- Typical search engines usually include invalid links and irrelevant web-pages through keyword inputs
- Need web-page classification for facilitating user searches
- Web-page classification is the primary requirement for search engines, which retrieve documents in response to the user query

# BACKGROUND

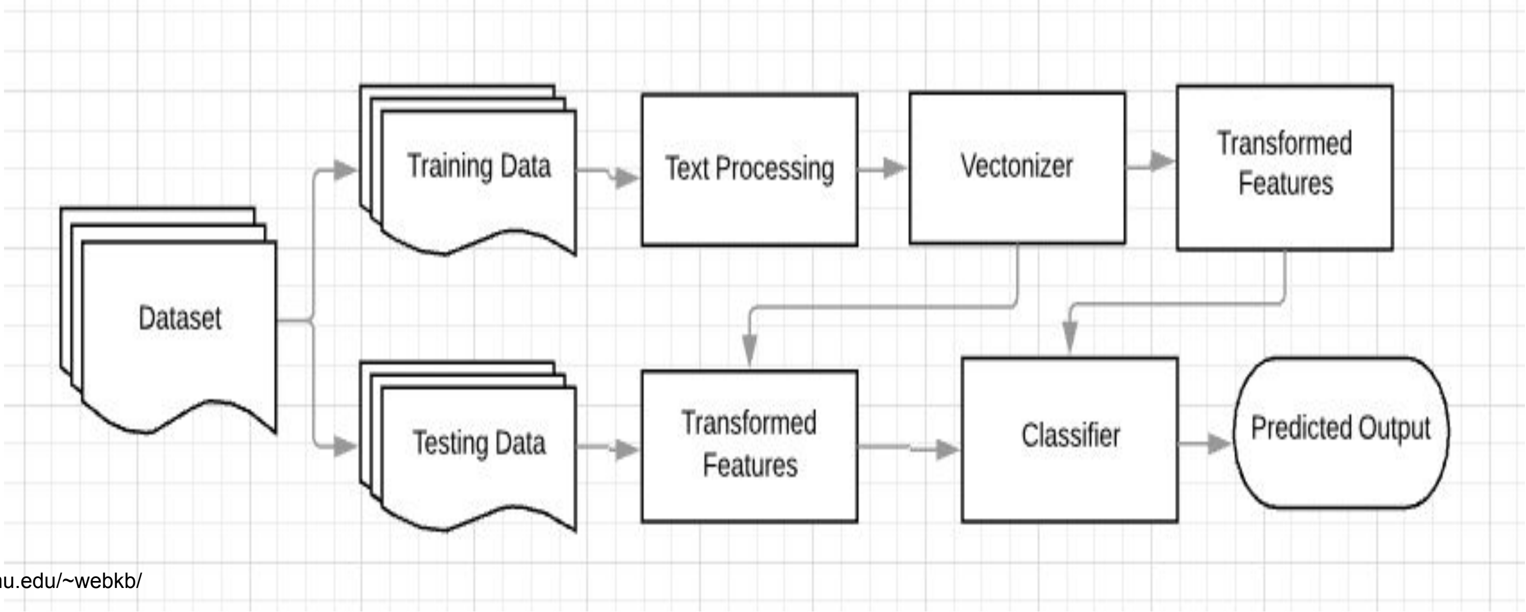
- The Web provides dynamically changing environment, which makes it difficult to build a classification model that can fit to classify different web pages.
- Many experiments like text based classification, link based classification have been done to enhance the efficiency of classification.
- After literature review, various studies show that linear SVM performs better for web page classification.
- Naïve Bayes being a probabilistic model also works well for text classification.
- Sequence models in Deep Learning such as LSTMs have been known to perform well in text-based classification.

# AGENDA

- Discussion about IntelliWeb Approach
- Classification Algorithms used:
  - Naïve Bayes with TF-IDF vectorization
  - Support Vector Machine with TF-IDF vectorization
  - GloVe based LSTM
- Evaluation results after 5 fold cross validation
- Conclusion
- Limitations
- Future Work

# WORKFLOW & DATASET (WebKb)

| **classes** | Student | Faculty | Staff | Department | Course | Project | Other |
|---|---|---|---|---|---|---|---|
| **# docs** | 1641 | 1124 | 137 | 182 | 930 | 504 | 3764 |




http://www.cs.cmu.edu/~webkb/

# TECH STACK

scikit learn

NumPy

matplotlib

K Keras

jupyter

Pandas

Natural Language ToolKit

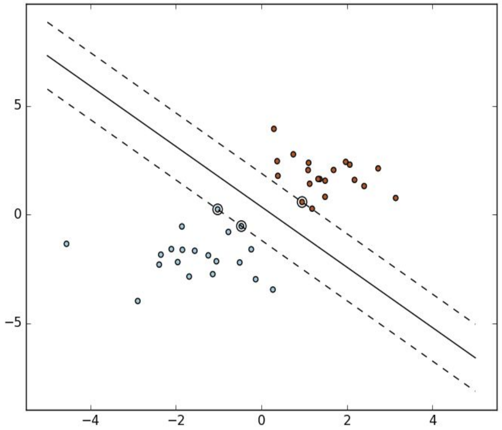
python™

# NAÏVE BAYES

- Naive Bayes Assumption
  - Conditional independence among the features
- Simple and Effective
  - Lower computation complexity for large number of features
- Uses probability of each attribute in each class
  - Frequency of words transformed to conditional probabilities
  - We use Multinomial Naive Bayes
- Two documents are said to be correlated if they belong to the same category specified by the conditional probabilities based on the frequency of word

# SUPPORT VECTOR MACHINE

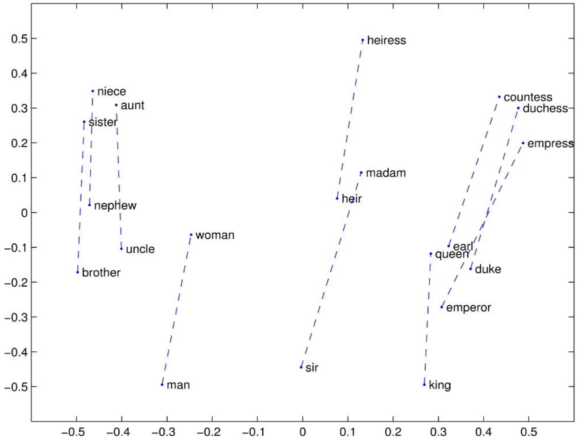
- Classify data using single or multiple hyperplanes.
- Less prone to overfitting.
- Can be trained in higher dimensions using "kernel trick".
- Computationally inexpensive compared to other classifiers.
- Input vector
  - Preprocessed data of web page content using NLTK.
  - TF-IDF Vectorization of preprocessed text.
- Performed Grid Search for SVM
  - Best parameters: C = 10 and kernel = ‘linear’.
- Various modifications of features tested with the inclusion of title of the webpage and hyperlinks.



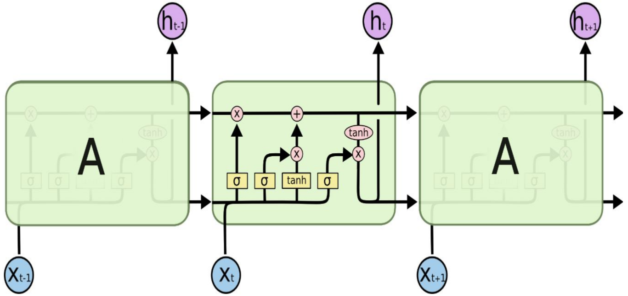

Source: https://scikit-learn.org

# GloVe and LSTM

- Global word-word co-occurrence matrix
- Log-bilinear model with a weighted least-squares objective
  - Word vector such that dot product equals the logarithm of the words' probability of co-occurrence
- Embed 'glove.6B.100d.txt' in our corpus
- LSTM - Long Short-term Memory
  - Special kind of Recurrent Neural Network, learn long term dependencies
  - Cell state regulated by structures called gates
  - Forget gate - Sigmoid layer to throw away information
  - Input gate - tanh layer to update cell state
  - Output gate - tanh layer used to filter what to output




Source: https://nlp.stanford.edu/projects/glove/




The repeating module in an LSTM contains four interacting layers.

Source: http://colah.github.io/posts/2015-08-Understanding-LSTMs/

# LSTM - Model Summary

```
_________________________________________________________________
Layer (type)                 Output Shape              Param #
=================================================================
embedding_3 (Embedding)      (None, 100, 100)          10000
_________________________________________________________________
lstm_5 (LSTM)                (None, 100, 128)          117248
_________________________________________________________________
dropout_5 (Dropout)          (None, 100, 128)          0
_________________________________________________________________
lstm_6 (LSTM)                (None, 128)               131584
_________________________________________________________________
dropout_6 (Dropout)          (None, 128)               0
_________________________________________________________________
dense_3 (Dense)              (None, 7)                 903
_________________________________________________________________
activation_3 (Activation)    (None, 7)                 0
=================================================================
Total params: 259,735
Trainable params: 249,735
Non-trainable params: 10,000
_________________________________________________________________
```

**Parameters:**

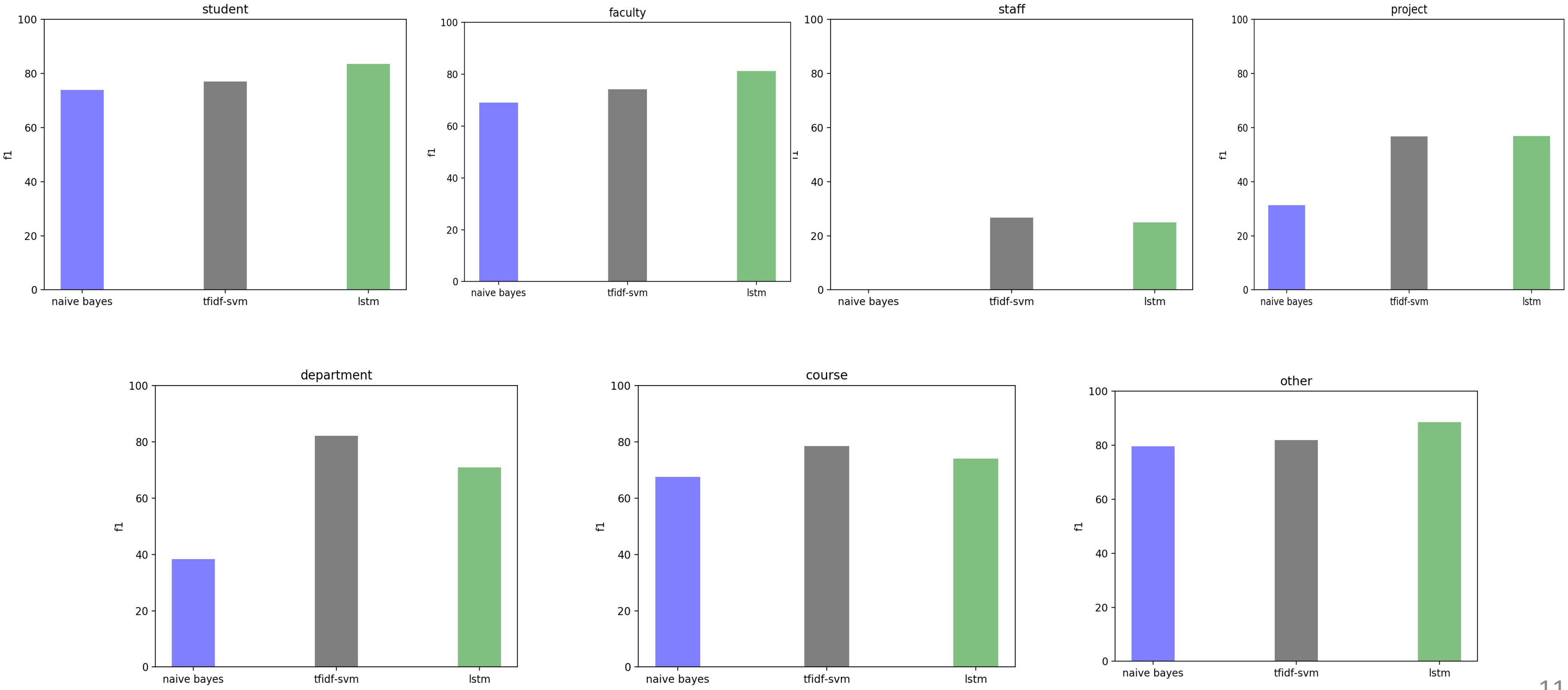
Glove:

- Embedding dim = 100
- Vocabulary size = 100
- Max length of document = 100

LSTM

- loss='categorical_crossentropy'
- optimizer='adam'
- batch_size = 1000
- epochs = 50
- validation_split = 0.1
- metrics=[metrics.mae, metrics.categorical_accuracy]

# RESULTS (F1-measure) - After 5 fold CV



# CONCLUSIONS

- On average, F1-score for LSTM model is better for most of the entities.
- Accuracy is highest for LSTM ( ~ 95% on average).
- Not enough data to train the classifiers in case of ‘staff’ and ‘department’.
- SVM always performs better than Naive Bayes. But LSTM is best.
- Pre-trained GloVe embeddings performs better than trainable Word Embeddings.
- SVM performs at par with LSTM as the dataset is not large enough for LSTM.

# LIMITATIONS

- Dataset size small for training LSTM.
- Computational resource availability to train deeper neural network.
- Exploration of ensemble methods.

# FUTURE WORK

- Establish relationship model between the entities
  - Use the relationship model for classification
- Hyper-parameter Optimization of LSTM
- Statistical tests for significance and effect size
- Train Deep Learning module on larger amount of data
- Exploration of other types of web page classification problem e.g. news websites

# REFERENCES

[1] Freitag, D.\ (1998) Information extraction from HTML: Application of a general machine learning approach. AAAI/IAAI}.

[2] Furnkranz, J., Mitchell, T.\& Riloff, E. (1998) A case study in using linguistic phrases for text categorization on the WWW. { *Working Notes of the AAAI/ICML*}, { *Workshop on Learning for Text Categorization*}.

[3] Shen, D., Chen, Z., Yang, Q., Zeng, H., Zhang, B., Lu, Y., Ma, W. (2004), Web-page classification through summarization. { *Proceedings of the 27th annual international ACM SIGIR 04 conference on Research and Development in Information Retrieval*}, New York, ACM Press, pp:242- 249.

[4] McCallum, A.\ \& Nigam, K.\ (1998) A Comparison of Event Models for Naive Bayes Text Classification {*AAAI Workshop*},{ *Workshop on Learning for Text Categorization*}