Personalised content recommendation for stack overflow

Karthik Seshadri,Nandini Rangaswamy

Agenda

- Overview
- N-gram based similarity search
- Random Forest
- SVM
- Evaluation of 3 solutions
- Results
- Conclusion and Future Work

Overview

- StackOverflow features a large corpus of knowledge and content covering many areas of Computer Science and Software development in particular
- Existing stackoverflow tools face 2 limitations
  - No user customizable recommendation system
  - Existing newsletter chooses content from all areas and only small subset is useful to user
- Purpose of the project is to develop personalized content recommendation system for stackoverflow

N-gram based similarity search

- Construct N-grams from the articles in stackoverflow
- Construct N-grams from the input user string
- Calculate the distance between the strings of the two N-gram sets
- Sort and display the top k posts with minimum distance

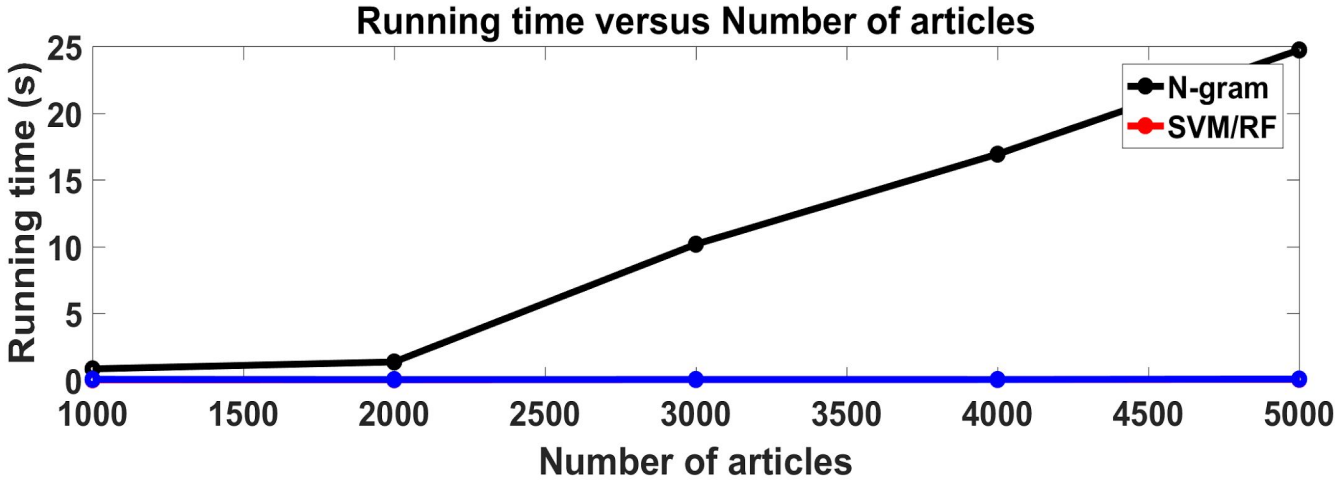
Random Forest

- Decision trees are trained on randomly chosen subset of articles
- Each split node chooses n-gram that maximises utility among randomly chosen subset of n-grams
- Generate Feature vector for the training set and test set which is bi-gram’s tf-idf
- Calculate the posterior probability based on fully trained classifier
- Sort and display top k posts with the highest probabilities

Support Vector machine

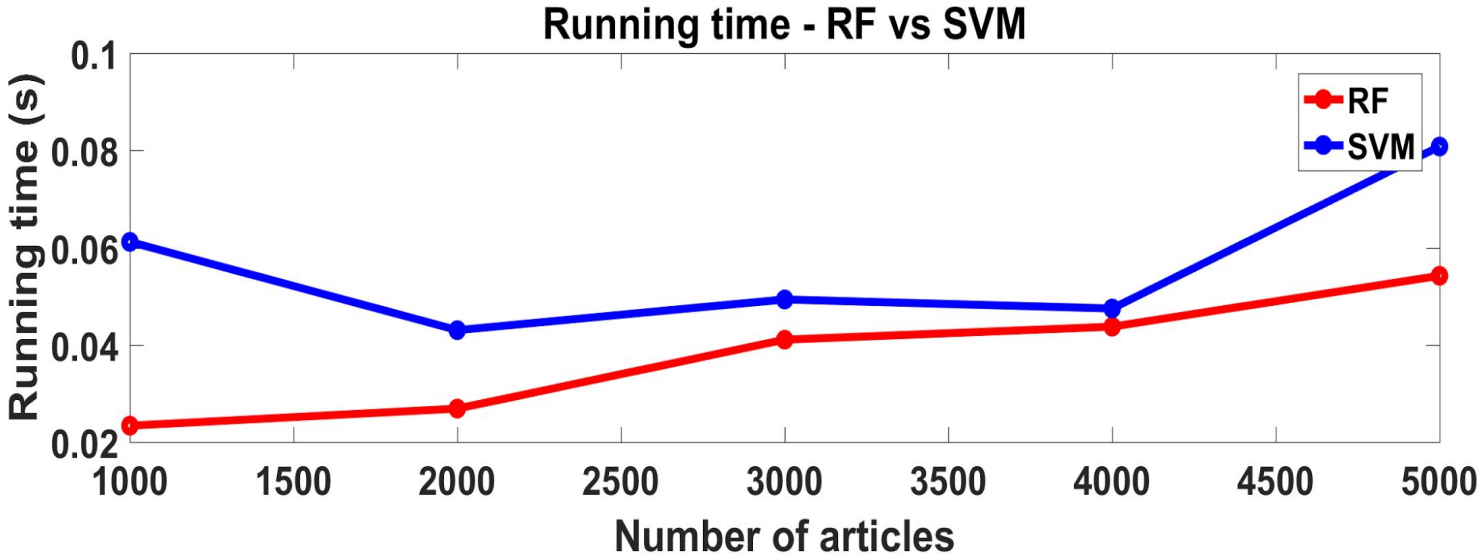
- SVMs learn a separation between positive and negative examples
- A simple linear SVM is used because the decision problem is binary
- Posterior probability is defined as distance of feature vector to hyper-plane

Evaluation criteria : Running time



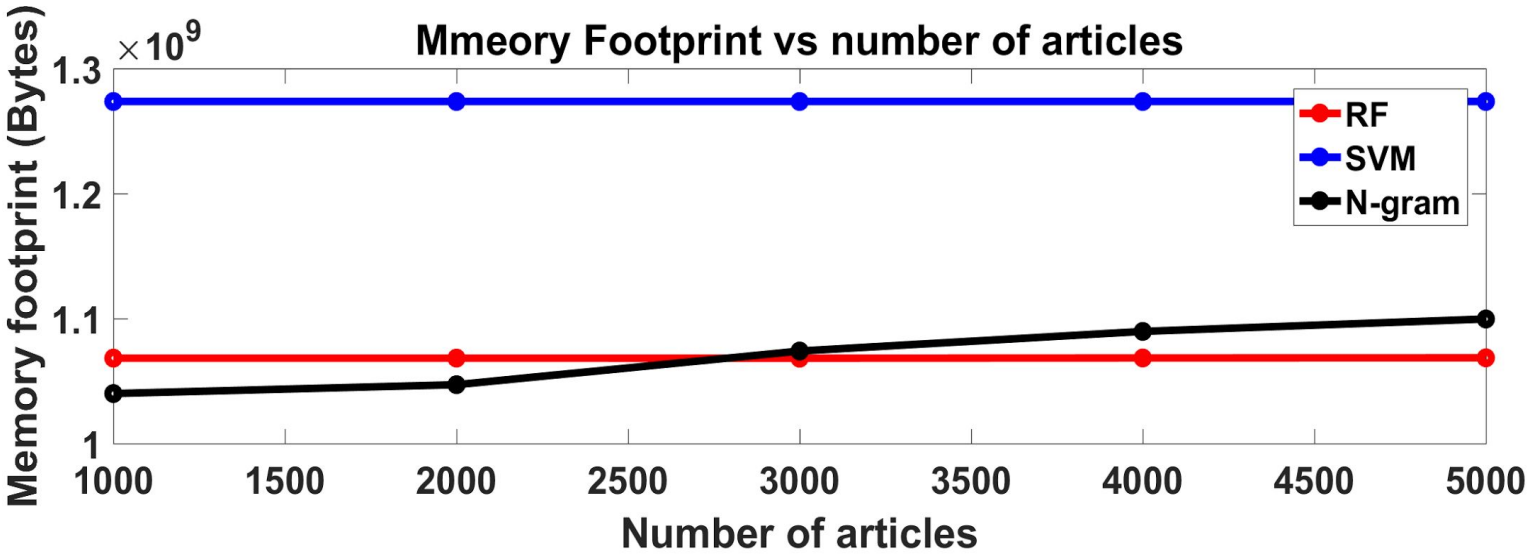
The n-gram approach has an order of magnitude higher run time that the machine learning based approaches

Evaluation criteria : Running time



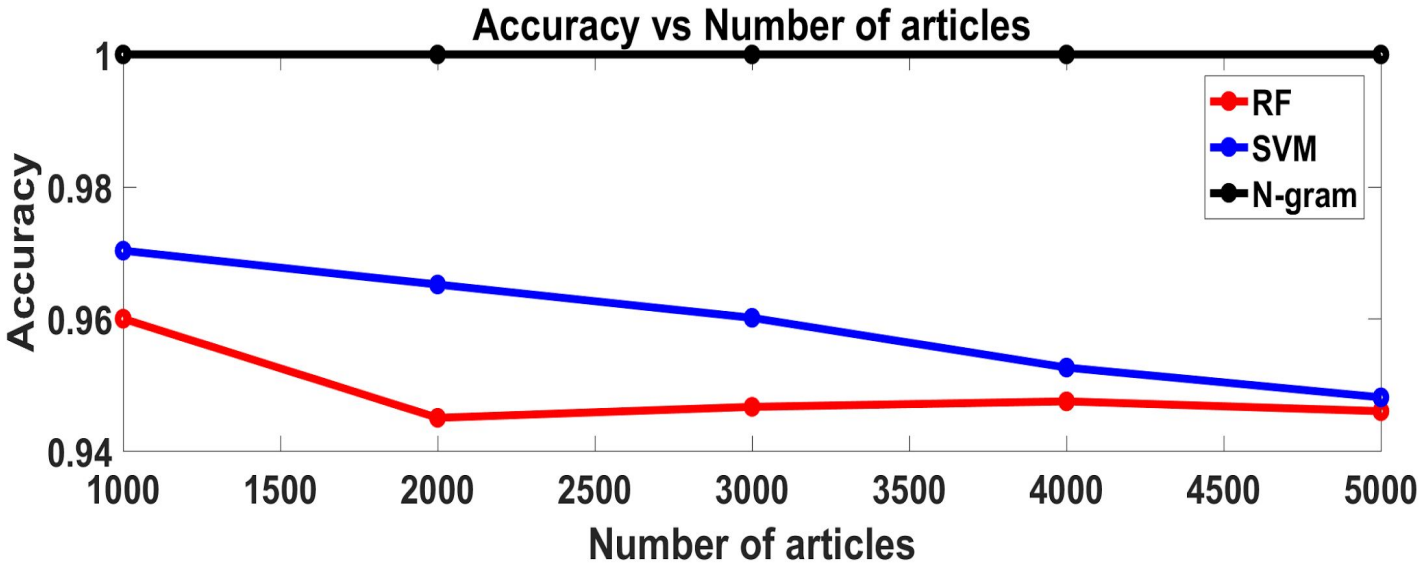
The majority of articles are processed in a timeframe less than or equal to 600 ms, less than a second. This therefore shows that both ML approaches can be used to carry out real time content recommendation in the collaborative filtering sense

Evaluation criteria: Memory



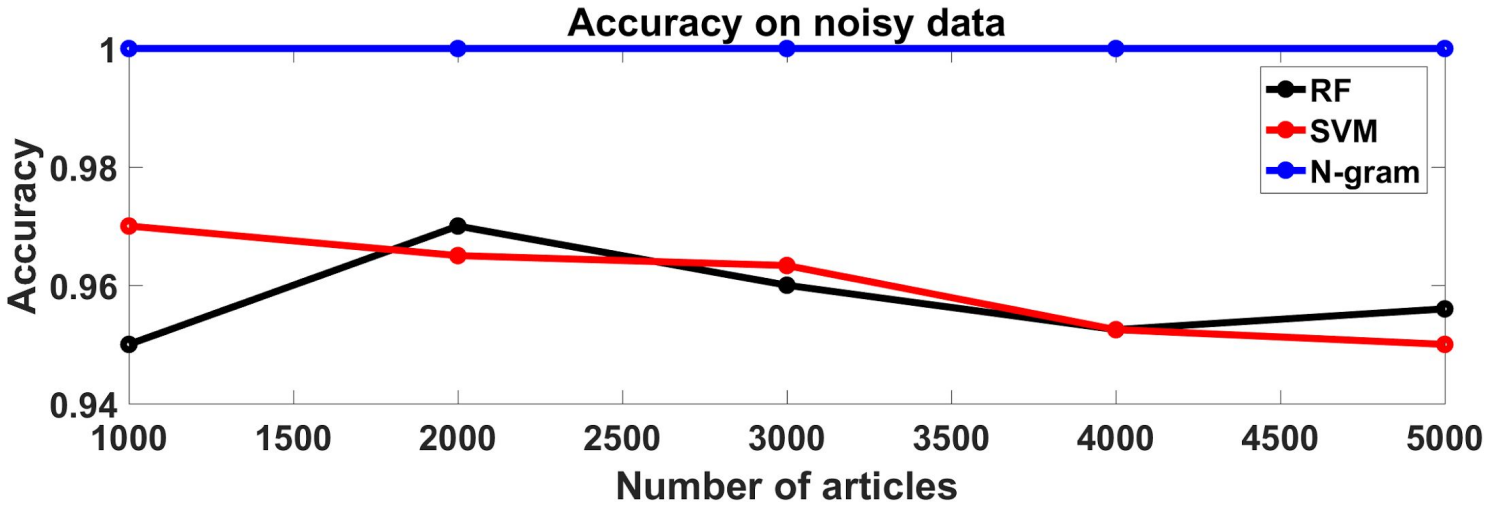
n-grams consumes much greater resources than the ML approaches

Evaluation criteria: Accuracy



N-gram has high accuracy than ML algorithms

Robustness to noise



Results

- N-gram approach has 100% accuracy
- Accuracy of SVM and random forest degrades with increase in data set but it is still above 94%
- N-gram approach has space complexity of O(n*n)
- There is not much difference between SVM and Random forest(RF is marginally lower than SVM) in terms of space complexity which is of the order of O(k) and both grow at a constant rate
- Running time of N-gram approach is of the order of O(n*n)
- Running time of SVM and RF is far lower compared to N-gram and RF is marginally faster compared to SVM
- N-gram has better robustness to noise than SVM or RF

Conclusion

- N-gram approach has too high a run time and memory footprint to be a feasible solution, atleast for real time applications.
- SVM and RF approaches performed very similarly in most areas, however the RF approach was marginally faster but less robust to noise for our particular application.
- We recommend Random Forest as the system of choice.