Smart Chat

Rishi Dua
Harvineet Singh

Real-time chat-based personalized recommendation system

Smart Chat

Motivation

- Combining Chat and Information
  - Limited features of traditional chat systems
  - No recommendations based on
    - User interests and needs
    - Schedule and planning
  - No application of NLP techniques for unstructured text (chat)
  - E-commerce websites: Amazon, Ebay etc.
  - Based on viewed and purchased products
  - More products suggested based on previous choice
- Personalisation with Privacy

Smart Chat

Problem Statement

- Developing a chat application providing real time suggestions
  - Movies: Show timings, recent movies
  - Food: Restaurant reviews
  - Products: Reviews, shopping links
  - Travel to city: Train/flight booking, Hotels, weather etc.
  - Plans/Meetings: Calendar and personal schedule
    And so on..choice
- Recommendations improved with user interaction
  - Learns preferences and interests
  - Recommendations personalised with user feedback
  - Tracking user clicks to improve classification accuracy

Smart Chat

Existing Work

- Applications developed
  - GaChat: Returns information from Wikipedia relevant to users chat
  - SemChat: Extracting Personal Information from Chat Conversations
- Current research
  - Topic detection and extraction in chat
  - Information extraction from chat corpus study
- Recommendations improved with user interaction
  - Trained on fixed given corpus
  - None of the research techniques give real-time recommendations

Smart Chat

Approach

- Live Recommendation System
- User sentences categorised into predefined categories: food, cars, movies, travel etc.
  - Using NLP, figure out the topic of conversation
  - Find relevant websites
  - Using open source search APIs to find recommendations from given websites
  - Users feedback used to improve future recommendations

User Interface

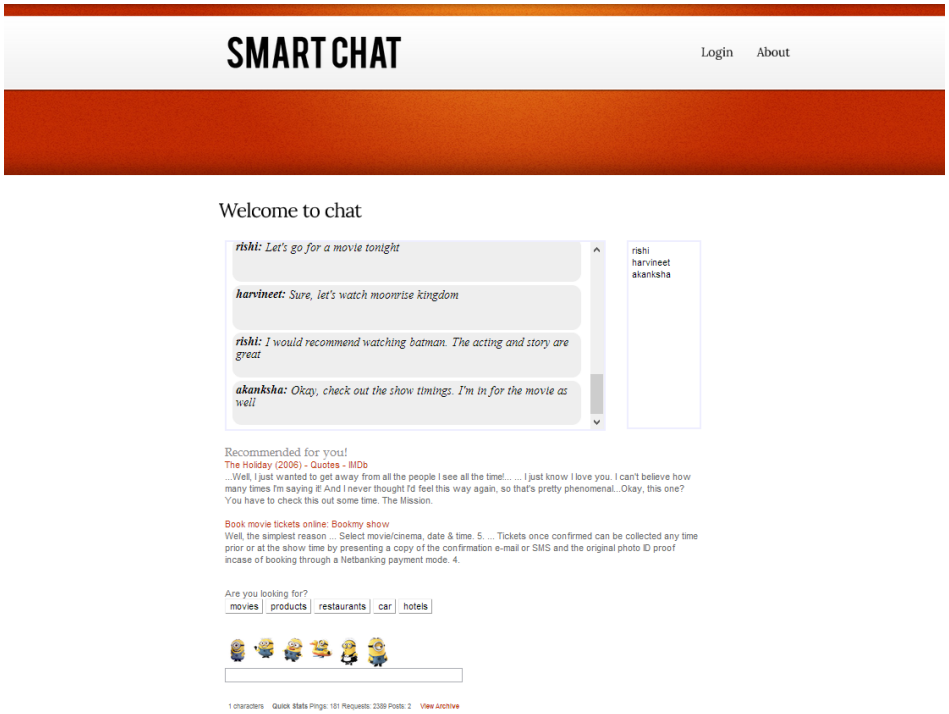

Figure: Screenshot of the User Interface

Smart Chat

Deciding when to give suggestion

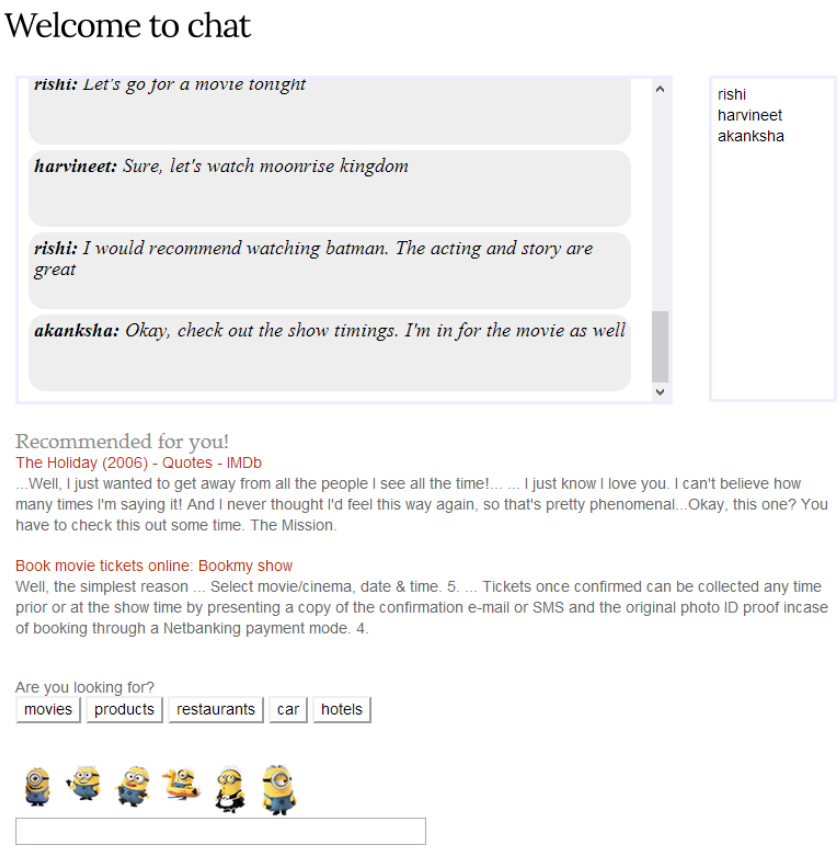

Figure: Recommendation is shown when 3 consecutive sentences are from same topic

Smart Chat

Feedback from user to improve prediction

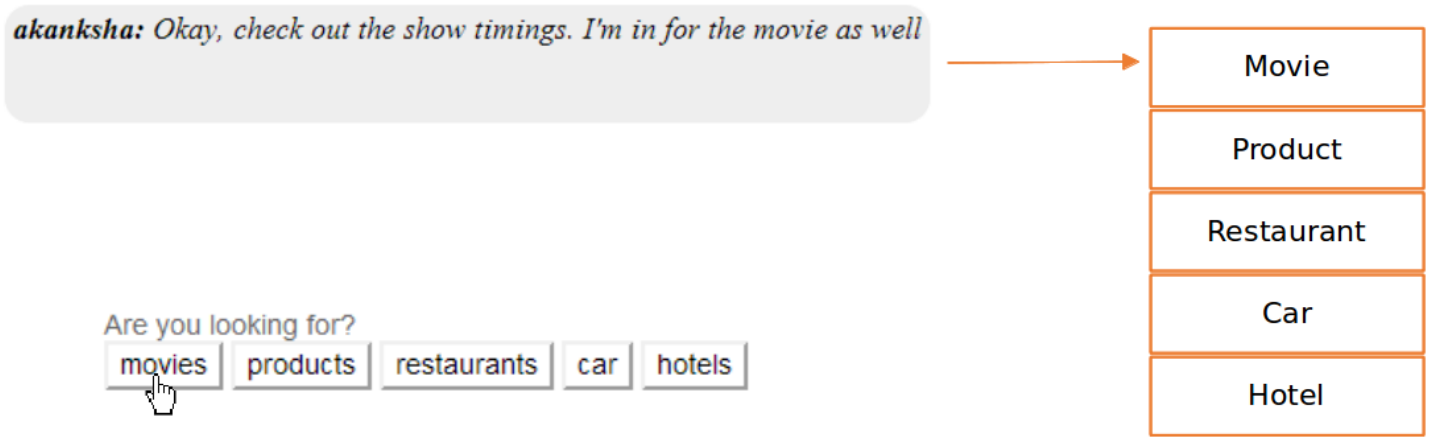

Figure: Screenshot of the User Interface

Smart Chat

Tracking user preference

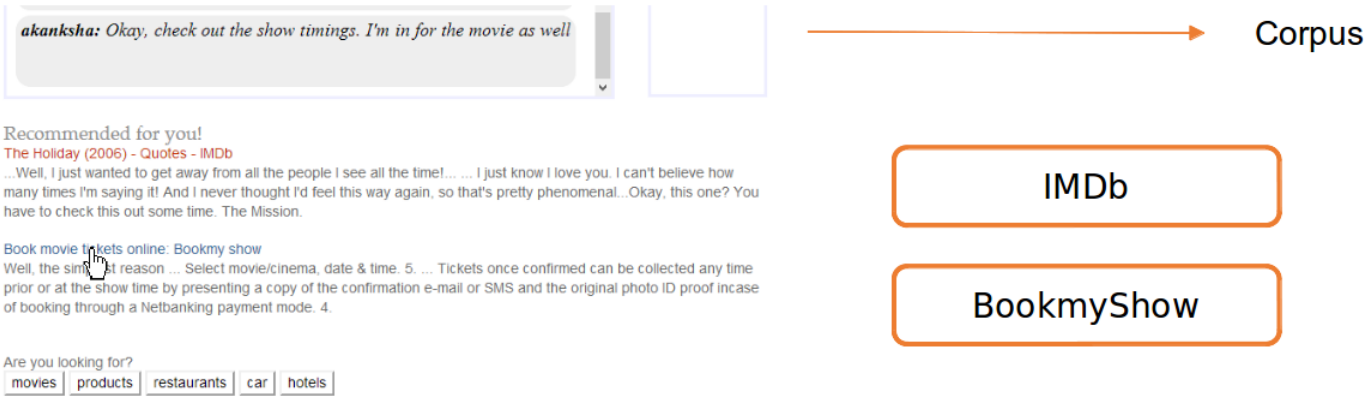

Figure: Ranking of recommendation gets updated when user clicks on a recommendation

Smart Chat

Privacy

- Chat data saved anonymously in corpus
- Training on chat session instead of users chat history

Smart Chat

Algorithm - Classification

1. Pre-processing: Stop words removal, Stemming using WorldNet lemmatizer
2. Feature extraction: TF-IDF of word count vectors
3. Feature selection: Select k-best features using Chi-square feature selection
4. Classification: Multinomial NB, Bernoulli, LinearSVC, Perceptron using scikit-learn library
5. Cross-validation: f-1 score to select best classifier

Smart Chat

Algorithm - Recommendations

1. Figure out the topic of conversation from learnt model
2. Mark chat as discussion if 3 consecutive sentences are from same topic
3. If chat is a discussion
    1. Search for recommendations from given websites using open source search APIs
    2. Track users clicks
    3. On clicking a recommendation:
        - Mark text as correctly classified and add to corpus
        - Increase score of the recommended website for future recommendations
    4. On clicking a feedback link
    5. Mark text as incorrectly classified and add to marked corpus

Results Selecting Algorithm

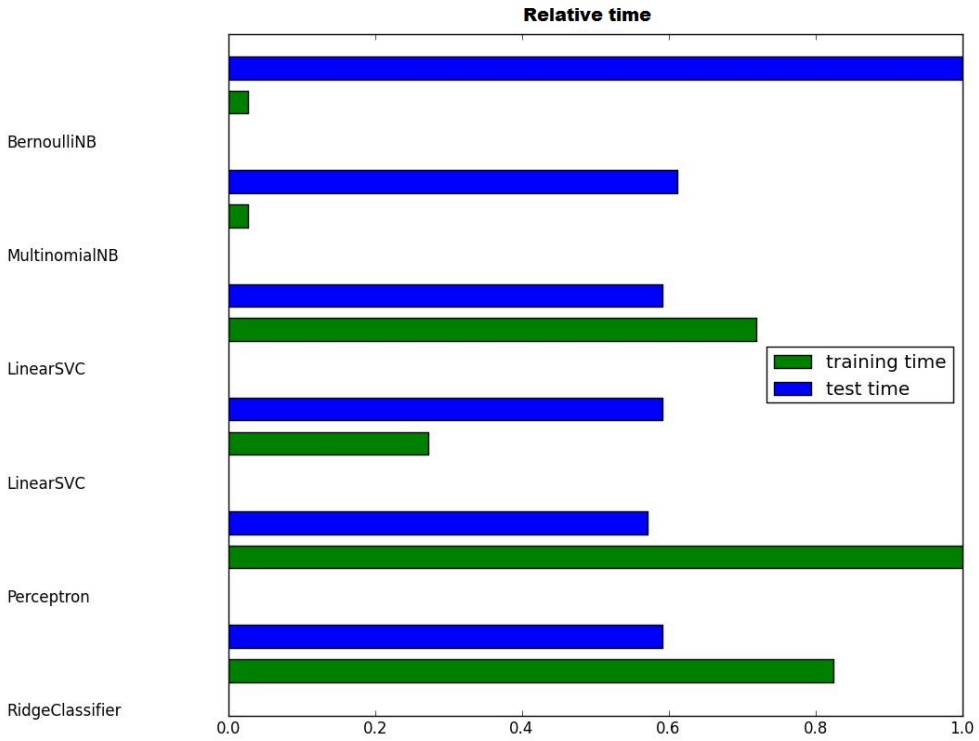

Figure: Comparison of learning algorithms on our dataset

Results Feature Selection

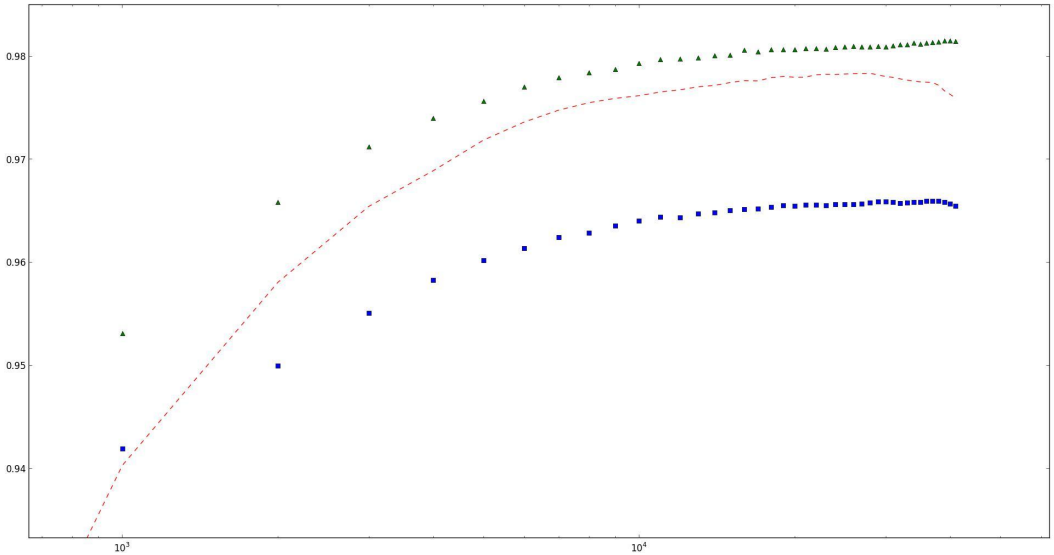

Figure: K best features (Log scale) using Chi-Square measure

Top: Linear SVM, middle: Multinomial, bottom: Bernoulli

Multinomial outperforms Bernoulli for large vocabulary
Good accuracy obtained on using 5000 words as features

Smart Chat

Future Scope

- Personalization of the feature extraction technique
- Clustering of users based on their preferences
- Maintaining an activity calendar to remind user of his schedule while planning tentative work on chat.
- Yet to implement the calendar feature in the application
- Changing from a Ajax based chat to an XMPP chat server
- Integrating with Facebook chat API