Home Depot Competition

Chirayu Wongchokprasitti, PhD

University of Pittsburgh

Center for Causal Discovery

Department of Biomedical Informatics

chw20@pitt.edu

http://www.pitt.edu/~chw20

Competition Detail

- Predict the relevance of search results
- End: 11:59 pm, Monday 25 April 2016 UTC (98 days left)
- Evaluation: Root Mean Squared Error (RMSE)
- Data: Train, Test, Product Descriptions, Attribute

  https://www.kaggle.com/dsoreo/home-depot-product-search-relevance/testing-r/notebook

Preparing Data

- Cleanup Data
  - Lower Case
  - Remove Special Characters (Remove White Space/Tab)
  - Remove Stop Words (Too Common Words/Terms)
  - Correct misspelled words (aircondition -> air condition)
- Stem Data (Disambiguating different forms of the same word)
  - Porter Stemmer
  - Snowball Stemmer
- Normalize Data
  - TF/IDF
  - Z score $(x - \mu)/\sigma$
- Feature Manipulation (Beyond Bag of Words)
  - Feature Expansion (N-Grams)
  - Feature Transformation (SVD)
  - Feature Selection (Forward Selection/Backward Elimination/PCA)

Correcting Misspelled Words

- Naiive Bayes
  - Filter the good word outs (Lookup a dictionary http://services.aonaware.com/DictService/ or https://www.wordsapi.com/)
  - Lookup Google N-Grams as reference (Find Frequency of any term) http://storage.googleapis.com/books/ngrams/books/datasetsv2.html
  - Generate all possible combination terms
  - Pick highest prob. one
  - Ex: P(air)*P(condition) > P(a)*P(ir)*P(con)*P(dit)*P(ion)
- Lean on Google Search (Search Suggestion) https://www.kaggle.com/steubk/home-depot-product-search-relevance/fixing-typos/discussion
  - (And it doesn't break a competition's rule ☺)
- http://hunspell.github.io/ (Comply the rule ☺)

Bag of Words (Unigram)

id	q_1	… q_m	title_1	… title_n	p_1	… p_o	a_1	… a_p
1	1	0	1	0	0	0	0	0
2	0	0	0	1	0	0	0	0
3	0	0	0	0	0	1	0	0
4	0	1	0	0	5	0	0	0
5	0	0	0	1	0	0	0	0
6	1	0	0	0	0	0	0	0
7	0	0	0	0	0	3	0	0
…								
N								

- Still need to normalize the values

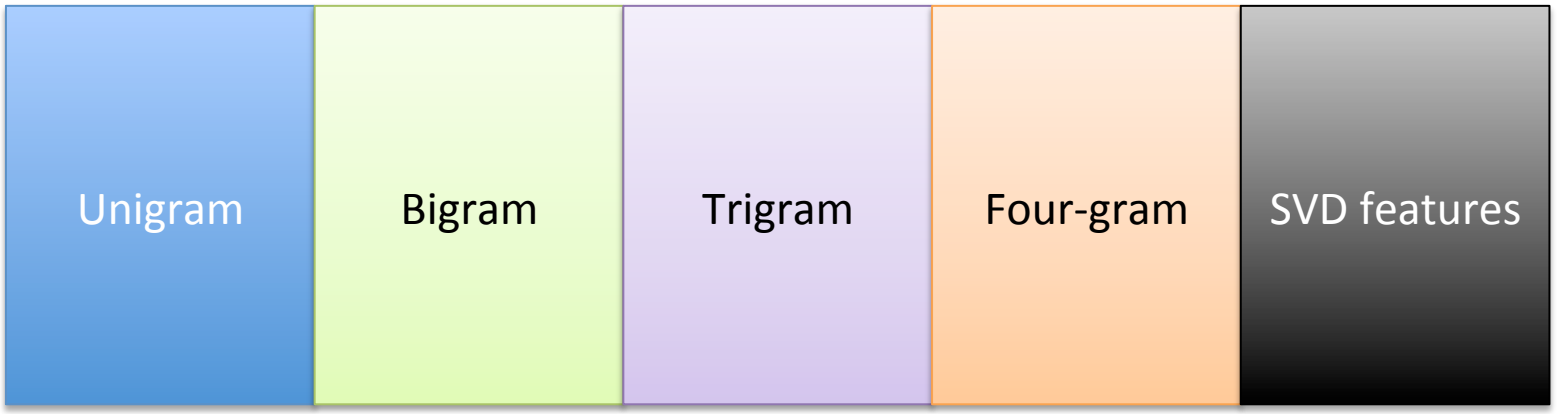
N-Grams

- A contiguous sequence of n items from a given sequence of text or speech
- The items can be phonemes, syllables, letters, words or base pairs

Feature Transformation

- Singular Value Decomposition
  - M = UΣV*
  - U is the interaction matrix btw features
  - V is the interaction matrix btw products
- Word2vec (i.e. king – man + woman = queen) http://deeplearning4j.org/word2vec
- Lda2vec https://github.com/cemoody/lda2vec and http://www.slideshare.net/ChristopherMoody3/word2vec-lda-and-introducing-a-new-hybrid-algorithm-lda2vec-57135994

Feature Space



Example

- Python
  - Random Forest (score w/o stemming: 0.49739) https://www.kaggle.com/wenxuanchen/home-depot-product-search-relevance/sklearn-random-forest
  - Random Forest (score w Snowball stemming: 0.48721) https://www.kaggle.com/junfeng/home-depot-product-search-relevance/sklearn-random-forest-merge-attributes/log
- R
  - (Boosting) https://www.kaggle.com/junfeng/home-depot-product-search-relevance/sklearn-random-forest-merge-attributes/code