Data Science Take-home Project

Eluvio ML Data Science/ML positions

Wen-Hao Chiang
01/15/2021

Statistics and Problem statements

- **Data Statistics**
  - Potential informative features to model with:
    - Titles, time_created, date_created, up_votes, and author
      - All down_votes are zero. All belong to ‘world_news’ category.
      - There are only 320 over_18 out of 509,236.

- **Research Problems:**
  - Popularity prediction
    - We aim to build a model to predict the # of up_votes for each post
  - Post clustering
    - We aim to analyze the posts based on the clustering property
  - Topic of post recommendation for authors
    - We aim to recommend authors post of interest based their previous post.

Popularity Prediction

- **Binary classification: Popular vs.. Non-relevant posts**
- **Popular vote labeling**
  - # of posts: 509,236 from 2008-01-25 to 2016-11-22
  - Quick overview for # of post per day Skewed.

	count	mean	std	min	25%	50%	75%	max
n_post	3223.0	158.000621	97.360446	1.0	74.0	129.0	241.0	458.0

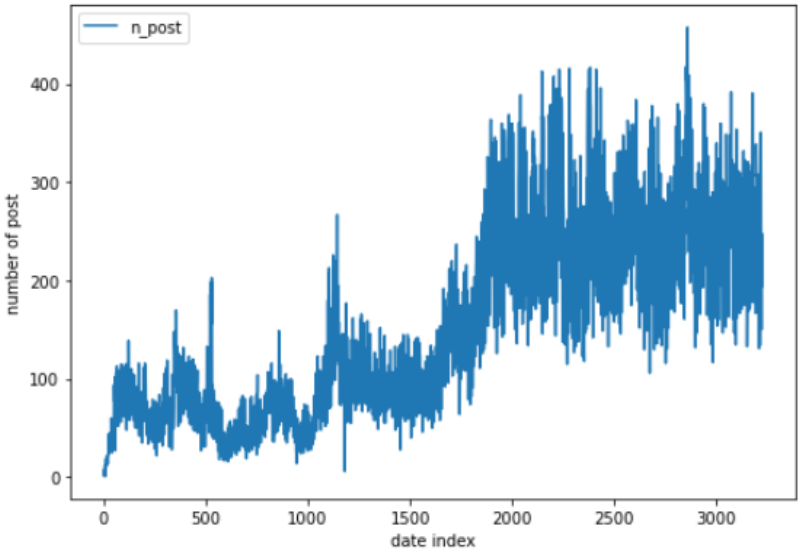
  - Average up_votes per day

	count	mean	std	min	25%	50%	75%	max
up_votes	3223.0	90.152178	65.953134	1.245902	35.178536	79.960396	128.165878	477.19084

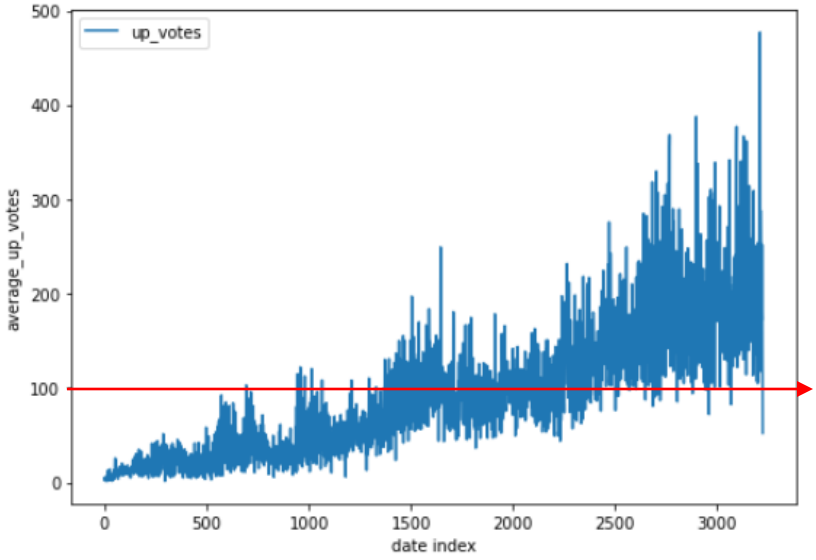
  - Can we use mean as a threshold to determine the popular news?
    - ◆No. The average up_votes per day is skewed.
    - ◆Different day has different up_votes range.
    - ◆Double check ?(next slides)

Popular labeling

Number of posts per day



Average up_votes per day

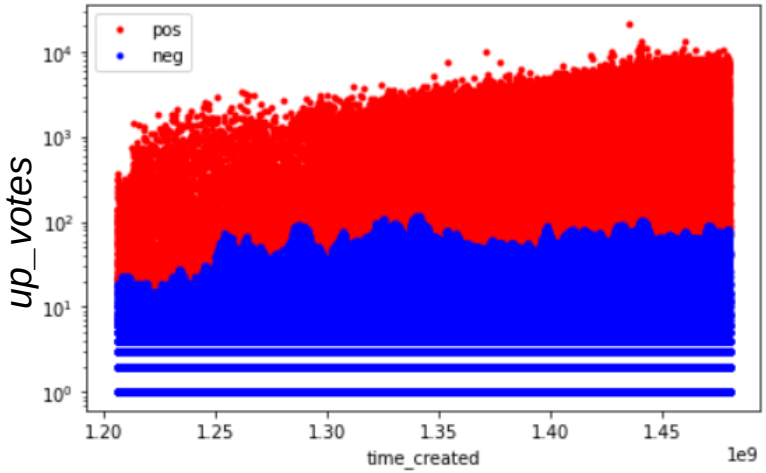


Threshold?
No.

- **Threshold can't be average up_votes.**
- **User were growing.**
- **The popularity should reflect the number of users.**
- **How to label relatively popular vote?**

Relative Popularity

- **A post should be considered popular:**
  - Compared to the previous month and the day of the week.
    - ◆(remove season and periodical trend effect)
    - ◆Bin the post and up_votes from previous month and the day of the week
  - A popular vote is a positive outlier inside that bin
  - Outlier criteria (Popular): mean + 3 * interquartile range (IQR)
  - The 3 is changeable and the difficulty of the classification problem is defined by the pattern intensity and confidence (i.g., 3) in the label.
  - Figure: upvotes and its labels
  - More reasonable.



Binary Classification Modeling

- **Approach 1: Bag of words (Tf-Idf)**
- **Text Pre-procssing:**
  - Contractions Mapping
  - Format words and remove unwanted characters
  - Remove stop words
  - Tokenize each word
  - Lemmatize each token
  - Stem words: reducing each word to its root or base
  - We present the examples for titles and bad word in the next slide
- **Other steps:**
  - Key words and subject preserve
  - Subjects are important to news title and can be target of interests

Example of Bag of words

Title	Processes Words
Rightist Gangs Murdering Trade Unionists in Colombia	rightist gang murder trade unionist
Protesters gathered around a mosque in the west of the Afghan capital after Friday prayers chanting death to Denmark , death to the Netherlands, death to America and death to Jews .	protest gather around west afghan prayer chant death jew
car bomb exploded Friday outside a police station in Spain s northern Rioja region	car bomb station northern region
MinnPost - Pent-up hatred unleashed in shocking Lhasa riots	pent unleash shock riot

Solve Scalability Issues

- **Dataset Imbalance:**
  - Popular vs Unpopular: 63,003 vs 441,714 (1:7)
  - Class weight considered during training
- **In total, there are 11,245 words in vocabulary**
- **K-fold cross validation and report the average metrics**
- **Scalability: Use SGD classifier to train in batch**
- **Dimensionality reduction: direct PCA fail due to high complexity**
- **Solution:**
  - Feature (words) selections by Chi-squared stats
    Compute chi-squared stats between each non-negative feature and class
  - Retain ¼ words and plus its PCA results
    further reduce to 1000 dimensions
  - Chi-squared stats K best words based on label and PCA consider the direction with high variance

Model Comparison Approaches (1/2)

- **Scalability: Simple SGD Logistic Classifier**
  - **Deep Learning fit_generator (Only read each batch from HDD at each iteration)**
  - SGDlogit: Simple SGD logistic regression
  - SGDlogitPca: Simple SGD logistic with PCA and ChiTest for feature reduction
  - NN_fit: deep neural network
  - NN_fitPca: deep neural network with PCA and ChiTest for feature reduction
  - NN_nnlm_embed: neural network with sentence embedding layer
  - NN_nnlm_pretrain: neural network with pretrained embedding
  - NN_gnews_pretrain: neural network with google news pretrained embedding

	Accuracy	Recall	Precision	F1
SGDlogit	**0.584**	**0.564**	**0.169**	**0.260**
SGDlogitPca	**0.588**	**0.524**	**0.178**	**0.264**
NN_fit	**0.592**	**0.598**	**0.171**	**0.266**
NN_fitPca	**0.608**	**0.628**	**0.178**	**0.278**
NN_nnlm_embed	**0.551**	**0.429**	**0.158**	**0.231**
NN_nnlm_pretrain	**0.608**	**0.631**	**0.179**	**0.278**
NN_gnews_pretrain	**0.592**	**0.599**	**0.172**	**0.267**

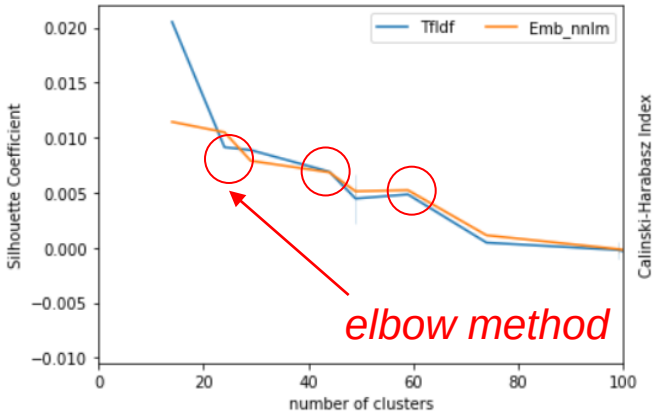
Model Comparison Approaches (2/2)

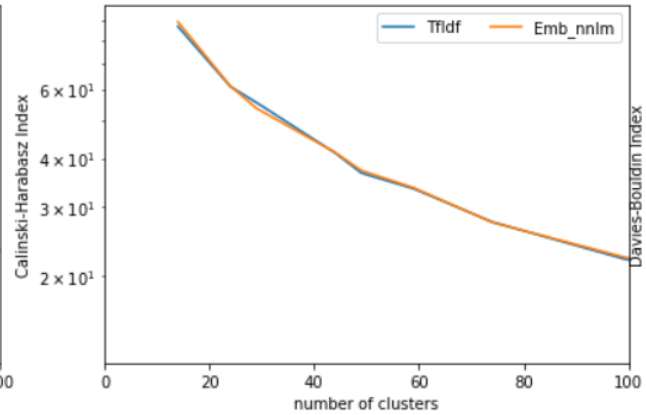
- **Balanced Accuracy: 60.8%**
- **Recall: 63.1%**
- **Precision: 17.9%**
- **F1: 27.8%**
- **The difficulty of the problem is defined by the labeling scheme. A better labeling scheme is required.**
- **Next: Clustering qualities based on Tf-Idf features**

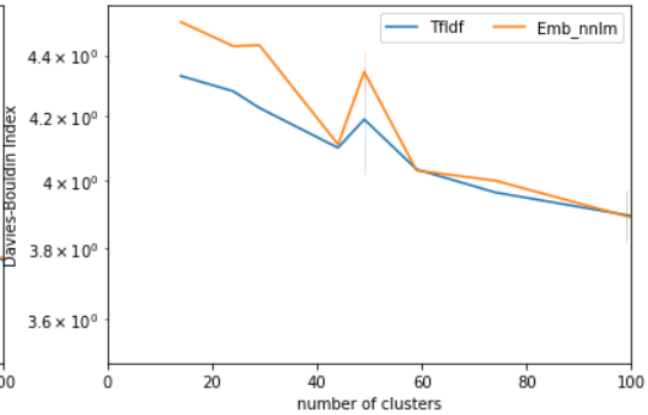
	Accuracy	Recall	Precision	F1
SGDlogit	**0.584**	**0.564**	**0.169**	**0.260**
SGDlogitPca	**0.588**	**0.524**	**0.178**	**0.264**
NN_fit	**0.592**	**0.598**	**0.171**	**0.266**
NN_fitPca	**0.608**	**0.628**	**0.178**	**0.278**
NN_nnlm_embed	**0.551**	**0.429**	**0.158**	**0.231**
NN_nnlm_pretrain	**0.608**	**0.631**	**0.179**	**0.278**
NN_gnews_pretrain	**0.592**	**0.599**	**0.172**	**0.267**

Clustering into Topic and Evaluation Metrics

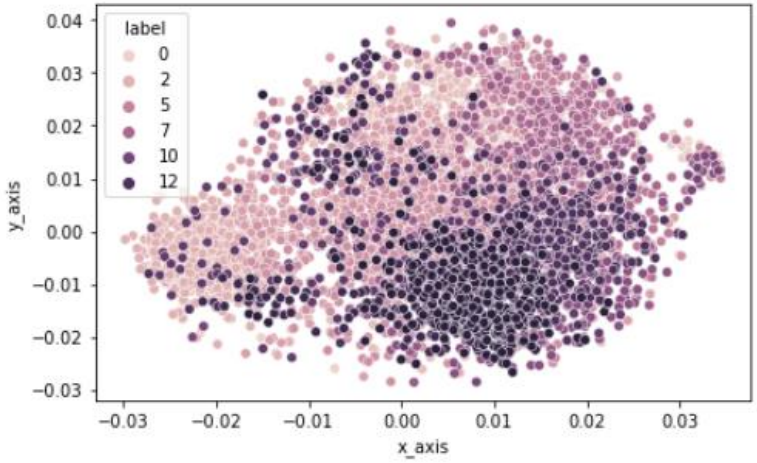
- Clustering qualities and choose the number of clusters
- Evaluation metrics: Silhouette Coefficient, Calinski-Harabasz, and Dunn index







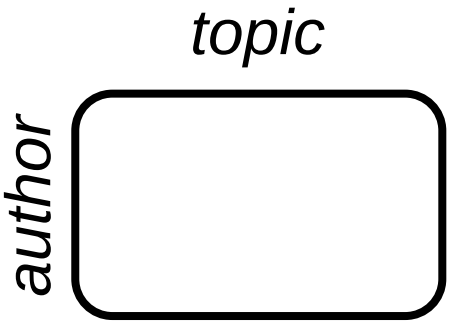
- TSNE: Clustering pattern
- Fig: Top-6 clusters
- Why clustering?
- Build topic recommendation based on common categories



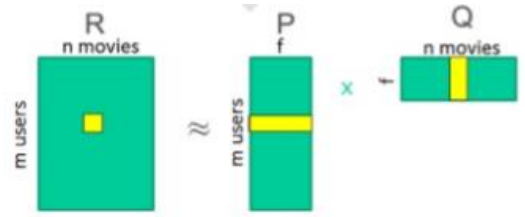
Topic (Clustering) and Post Recommendation

- **Define Recommendation Problem:**
  - Recommend posts of interest to the authors
  - Ground Truth: Hold out one post from its author as a post of interest to the author
- **Methodology: Clustering posts into different categories (topic of posts)**
- **Recommendation system model based on authors-topic matrix**
- **Recommend post from the model based on K nearest posts.**

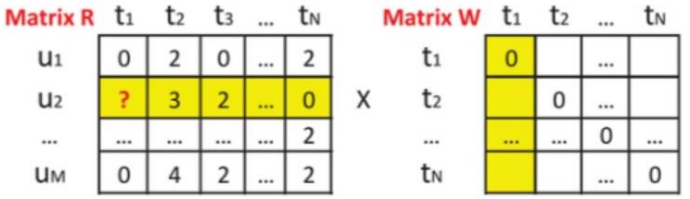
User-Topic Matrix



Matrix Factorization



Sparse LInear Method



Matrix R = rating matrix; W = coefficient matrix

$$\widehat{S}_{i,j} = R_{i,:}W_{:,j} = \sum_{\substack{h=1 \\ h \neq j}}^{N} R_{i,h}W_{h,j}$$

Experimental Setting

- **Clustering number is decided through the elbow method from silhouette coefficient : n=45**
  - Each cluster is considered as a topic
  - Retain authors with more than 10 topics to construct a dataset (3,987 authors)
  - Testing set: hold out one post from each topic for each author
- **Top-N Recommendation model:**
  - SLIM: Sparse LInear Method
  - Matrix Factorization
- **Evaluation: Hit rate@K, ARHR**

- Hit Rate (HR)

$$HR = \frac{\#hits}{\#users}$$

- Average Reciprocal Hit Rank (ARHR)

$$ARHR = \frac{1}{\#users} \sum_{i=1}^{\#hits} \frac{1}{pos_i}$$

Recommendation Results

- **Hold out category recommendation**

	HitRate@3	HitRate@5	HitRate@10	ARHR@3	ARHR@5	ARHR@10
SLIM	46.75%	57.06%	72.33%	0.3628	0.3864	0.4067
MF	32.02%	44.04%	63.90%	0.2328	0.2600	0.2863

- **Hold out topics from each author's previous post and do topic recommendation**
- **Given previous topic of interest, we can successfully recommend the topics of interest (from hold out posts) for users.**
- **SLIM has better results compared to Matric factorization**

Conclusion

1. Formulate popular post prediction problem as a binary classification
2. Compare classification performances including pretrained embedding and tf-Idf
3. Use chi-test and PCA for feature selection and dimensionality reduction
4. Evaluate topic clustering qualities
5. Use fit-generator for Neural Network and SGD classifier to tackle scalability issues
6. Build topic recommendation models based on the hold-out posts for each author
7. Evaluate performance for classification, clustering, recommendation.