Follow the hashtag: Popping the filter bubble

[Final Project Report]

Sweta Agrawal
130101089
IIT Guwahati
s.agrawal@iitg.ernet.in

Jitendra Choudhary
130101017
IIT Guwahati
c.babulal@iitg.ernet.in

Nikhil Agarwal
130101054
IIT Guwahati
c.babulal@iitg.ernet.in

Ravi Kumar
130101064
IIT Guwahati
ravi2013@iitg.ernet.in

ABSTRACT

With the increasing popularity of microblogging sites, we are in the era of information explosion. Although Twitter provides a list of most popular topics people tweet about known as Trending Topics, it is often hard to understand what these trending topics are about and what different opinions do people have about a particular topic. Since the number of tweets associated with a trending topic could be large, users generally get to see some selected tweets which is filtered through a mechanism where user select who to follow, and only see what they share. This Filter Bubble[6] undermine the marketplace of ideas and damage the ability to accurately predict events in the world. If user is never exposed to opinions or arguments that challenge his point of view, he will never question it or evolve it.

We propose a solution to the problem by providing the user a wider/global view of a particular trending event through exposing him to the available opinions world-wide. We evaluate existing techniques for encoding tweets into appropriate vector representations and discuss the pros and cons.

The dataset consists of tweets from Web Summit 2015 [1], US presidential election 2012 [2] and the FA cup [3].

1. INTRODUCTION

While social media have become mainstream, as shown by the evergrowing number of users, Twitter stands out as the quintessential platform to openly access real-time updates on breaking news and ongoing events. One of the appealing phenomena of the microblogging service is the fact that certain events produce a sudden increase of tweets in real-time as they unfold. The number of tweets in a trending event ranges from 1.2k to 100k. Thus filtering mechanism of the social media might lead to the situation in which the user receives biased information. In case of political information, it might lead to the situation that the user never sees contrasting viewpoints on a political or moral issue since he follows only like-minded people.

After this book[6] was published, researches have come up with different tools and techniques to fight the filter bubble. The chrome extension by MIT called FlipFeed[4] infers the political beliefs based on the users feed and then shows the feed of someone with opposing views. The major pitfall of this plugin is that the information gap is not significantly reduced.

Another tool named RightRelevance[5] aims at providing curated information and intelligence which includes

- Topic relationships including related topics & semantic information like synonyms, acronyms.
- Topical influencers (2.5M) with score and rank.
- Topical content and information in the form of articles, videos and conversations.

They use latent Dirichlet allocation (LDA) based text analysis of the tweets for identifying high value trending terms. They do a graphical analysis of twitter connectivity data to give interesting insights about an influential leader emerging as a result of a trend.

In our solution, we provide a simple tool that analyzes a real time trending event to extract opinions of people using clustering. Two challenging issues are notable in tweet clustering. Firstly, the sparse data problem is serious since no tweet can be longer than 140 characters. Secondly,synonymy and polysemy are rather common because users intend to present a unique meaning with a great number of manners in tweets. Also, Real time opinion mining of a particular trending topic can be difficult as tweets contains sarcasm, humor, anger, criticism and many different emotions. It has been argued that the Internet and social media increase the number of available viewpoints, perspectives, ideas and opinions available, leading to a very diverse pool of information. The main challenge is to be able to encode the tweet into a vector that understands opinion.

The paper [15] was one of the first to give an operational definition of redundancy. They define two tweets as redundant if the tweets either convey the same information (paraphrase) or if the information of one tweet subsumes the information of the other (textual entailment). Their have been several efforts to encode the tweet into a representation that identifies paraphrases. The paper[4] describe

[1]https://s3.amazonaws.com/aylien-main/misc/blog/data/websummit_dump_20151106155110.bz2
[2]https://datahub.io/dataset/twitter-2012-presidential-election
[3]http://www.socialsensor.eu/results/datasets
[4]https://www.media.mit.edu/projects/flipfeed/overview/
[5]http://www.rightrelevance.com/

an unsupervised approach for learning of a generic, distributed sentence representation. they evaluate their model on multiple tasks such as semantic relatedness,paraphrase detection, image-sentence ranking, question-type classification and benchmark sentiment and subjectivity datasets. In another paper[3], the author proposes a character composition model, tweet2vec, which finds vector space representations of whole tweets by learning complex, non-local dependencies in character sequences. The paper[13] present a novel method for generating general purpose vector representation of tweets using CNN-LSTM encoder decoder. They evaluate their model on multiple tasks such as paraphrase detection and sentiment classification.

There has been other several efforts in learning the word vector representation specific to Twitter Platform. The paper[10] describes learning sentiment specific word embedding (SSWE), which encodes sentiment information in the continuous representation of words. Specifically, they develop three neural networks to effectively incorporate the supervision from sentiment polarity of text (e.g. sentences or tweets) in their loss functions. GloVe[7] also provides pretrained word representation for tweets. The paper [9],[1],[8] describes experiments concerned with the automatic analysis of emotions in text such as anger, disgust, fear, joy, sadness and surprise. The last paper is specific to the context of twitter. In another paper[11],the author propose an emotional aware clustering approach which performs sentiment analysis of users tweets on the basis of an emotional dictionary and groups tweets according to the degree they express a specific set of emotions.

2. THE PROPOSED METHOD

The following flow chart summarizes the overall framework:

- Tweet Corpus: We use the paraphrase corpus provided as a part of SemEval 2015[6]. Also, to evaluate the existing models we use the dataset from Web Summit 2015 , US presidential election 2012 and the FA cup.
- Preprocessing: There are tools built specifically for preprocessing twitter dataset[7]. This library makes it easy to clean, parse or tokenize the tweets. URLs, mentions and hashtags are extracted.
- Opinion Vector Representation: We use various existing methods to extract the vector representation. These vectors are then projected onto a 2d space using t-SNE. t-distributed stochastic neighbor embedding[5] is a machine learning algorithm for dimensionality reduction. It is used for embedding high-dimensional data onto a 2D or 3D space.

3. TWEET REPRESENTATION

We compare various existing methods to create distributed representation of tweets.

3.1 K Means Clustering with Tf-idf Weights

The Tf-idf weighting ranks the importance of a term in a document. We represent each tweet using the tf-idf values of the terms. Then K-means clustering algorithm is applied to the vectors to generate clusters. K-means is a popular clustering algorithm that serves as dividing data samples into clusters, in which each observation belongs to the cluster with the nearest mean.

3.2 Singular valued Decomposition with tf-idf weights

Latent Semantic Analysis is also a technique for creating a vector representation of a document. The feature space created by tf-idf can be quite large and impractical to handle. Singular valued Decomposition is generally used for dimensionality reduction in unsupervised text learning problem.

3.3 Latent Dirichlet Allocation

LDA[2] is a generative probabilistic topic model that assumes documents as a mixture of topics and that each word in the document is attributable to the document's topics. In the context of text modeling, the topic probabilities provide an explicit representation of a document.

3.4 GloVe vectors

GloVe[7] is an unsupervised learning algorithm for obtaining vector representations for words. GloVe pretrained word vectors are used to get a representation for the tweet. The word vectors are concatenated to form the document representation.

3.5 Skip-thought vectors

The Skip-thought vectors[4] are trained in an encoder-decoder model that tries to construct the surrounding sentences of an encoded passage, i.e. for consecutive sentences S_{i-1}, S_i, S_{i+1} in some document, it predict target sentences S_{i1} and S_{i+1} given source sentence S_i.

The input sentence is first encoded by RNN and then decoded into two sentences. The decoder computes a softmax over the model's vocabulary. They evaluate their model for multiple tasks including paraphrase detection and semantic relatedness and achieve state-of-the-art results on both tasks. The cost of a training example is the sum of the negative log-likelihood of each correct word in the target sentences S_{i1} and S_{i+1}.

3.6 Tweet2Vec vectors

A Bi-directional Gated Recurrent Unit neural network is used for learning tweet representations. The input tweet is broken into a stream of characters and are represented by their one-hot vectors. The encoder consists of a forward-GRU and a backward-GRU, both having the same architecture, except the backward-GRU processes the sequence in reverse order. Finally, the tweet embedding learned by the encoder is passed through a linear layer whose output is the same size as the number of hashtags in the data set. The model[3] is also used by the paper[12] to cluster tweets based on the representation of the vector given by the model.

4. EXPERIMENTS

In this section we present an evaluation of the different distributed tweet representation.

4.1 Dataset

A Twitter Paraphrase Corpus[14] of about 18,000 sentence pairs, is used to evaluate existing models for paraphrase detection. Each sentence pair was annotated by 5 different

[6]http://alt.qcri.org/semeval2015/task1/

[7]https://pypi.python.org/pypi/tweet-preprocessor/0.5.0

crowd-sourcing workers. Since paraphrases are a subsets of similar opinion, the model needs to have a good performance for identifying paraphrases.

We evaluate the clusters manually by extracting the representations from all the methods described in section 3. A detailed discussion is presented in section 5. Our experimentation setup on each task is as follows:

- Use the pre-trained model for extracting the vector representation for the tweet
- Apply t-SNE to get the 2D projection of the embedded vector space.

5. RESULTS AND DISCUSSION

5.1 Paraphrase detection

The first evaluation is based on the SemEval 2015-Task 1 [14] Given two sentences, the task is to determine whether they express the same or very similar meaning. Acoording to the paper[15], paraphrases and entailments relates two tweets. Since, we want to cluster similar opinions, the paraphrases should belong to the same cluster.The ability to identify paraphrase, i.e. alternative expressions of the same meaning, and the degree of semantic similarity has proven useful for a wide variety of natural language processing applications.

We first extract the vector representation from skip-thought[4] model and tweet2vec[3]. Given two tweet vectors r and s, we compute the absolute component-wise distance(feature 1) between the two vectors and then train a logistic regression model with an L1 regularization on these features using the dataset. Other features were also used such as their element-wise product $r \cdot s$ and their absolute difference $|rs|$ with L2 regularization (feature 2) as mentioned in the paper[4]. Table 1 shows the performance of the model.

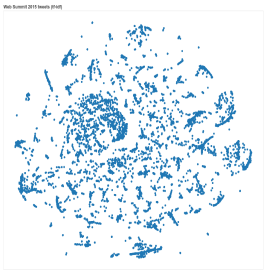
Model	feature	precision	recall	f1_score
skip-thought	1	0.72	0.73	0.72
skip-thought	2	0.65	0.67	0.65
tweet2vec	1	0.57	0.56	0.57
tweet2vec	2	0.60	0.59	0.59

Table 1: Results of the paraphrase detection in Twitter

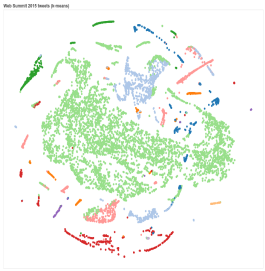
5.2 Clustering

The cluster formed by all the datasets from the representation embedded by all the models are provided in figure 2. Some of the interesting observations:

- In the first method, no rigid clusters are obtained. These are merely formed around keyword and not concepts.
- The clusters in K-means are again around keywords. However, the separation is improved.
- The topics identified by LDA incorporates hidden/latent concepts. The graph brings both similar texts and concepts together.
- The tweets in the forth cluster are a lot semantically closer than what is obtained from LDA.



(a) SVD with Tf-Idf weighting



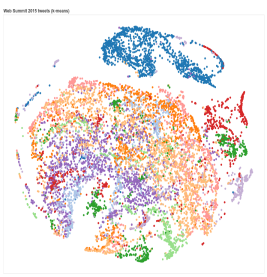
(b) K-Means with Tf-Idf weighting

(c) Latent Dirichlet Allocation

(d) GloVe Vectors

(e) Skip Thoughts

Figure 1: Web Summit Results



(a) Tweet2Vec

Figure 2: Web Summit Results

- The representation obtained from skip-thoughts were very interesting. The clusters represent opportunity, excitation, etc. However, they do not encode topical information.

6. CONCLUSION AND FUTURE WORK

In this paper, we used various existing methods to embed tweets into distributive representation. However, the work was not exhaustive. We identified the pros and cons of existing method. There are a couple of models that we didn't try in the current evaluation, tweet2vec[13], skip-thoughts trained on tweets dataset and hybrid methods that include best features from tweet2vec and skip-thoughts. We would like to include twitter graph data in future to evaluate generated clusters.

7. CONTRIBUTION

Jitendra Choudhary: I did most of the literature review and data collection/preprocessing.

Ravi Kumar & Nikhil Agarwal: Our major contribution was towards evaluating existing methods(the first four) on multiple dataset.

Sweta Agrawal: My major contribution was towards evaluating the skip-gram and tweet2vec model for paraphrase and clustering purpose.

8. REFERENCES

[1] S. Aman and S. Szpakowicz. Identifying expressions of emotion in text. In *International Conference on Text, Speech and Dialogue*, pages 196–205. Springer, 2007.

[2] D. M. Blei, A. Y. Ng, and M. I. Jordan. Latent dirichlet allocation. *Journal of machine Learning research*, 3(Jan):993–1022, 2003.

[3] B. Dhingra, Z. Zhou, D. Fitzpatrick, M. Muehl, and W. W. Cohen. Tweet2vec: Character-based distributed representations for social media. *arXiv preprint arXiv:1605.03481*, 2016.

[4] R. Kiros, Y. Zhu, R. R. Salakhutdinov, R. Zemel, R. Urtasun, A. Torralba, and S. Fidler. Skip-thought vectors. In *Advances in neural information processing systems*, pages 3294–3302, 2015.

[5] L. v. d. Maaten and G. Hinton. Visualizing data using t-sne. *Journal of Machine Learning Research*, 9(Nov):2579–2605, 2008.

[6] E. Pariser. *The filter bubble: What the Internet is hiding from you*. Penguin UK, 2011.

[7] J. Pennington, R. Socher, and C. D. Manning. Glove: Global vectors for word representation. In *Empirical Methods in Natural Language Processing (EMNLP)*, pages 1532–1543, 2014.

[8] K. Roberts, M. A. Roach, J. Johnson, J. Guthrie, and S. M. Harabagiu. Empatweet: Annotating and detecting emotions on twitter. In *LREC*, pages 3806–3813. Citeseer, 2012.

[9] C. Strapparava and R. Mihalcea. Learning to identify emotions in text. In *Proceedings of the 2008 ACM symposium on Applied computing*, pages 1556–1560. ACM, 2008.

[10] D. Tang, F. Wei, N. Yang, M. Zhou, T. Liu, and B. Qin. Learning sentiment-specific word embedding for twitter sentiment classification. In *ACL (1)*, pages 1555–1565, 2014.

[11] K. Tsagkalidou, V. Koutsonikola, A. Vakali, and K. Kafetsios. Emotional aware clustering on micro-blogging sources. In *International Conference on Affective Computing and Intelligent Interaction*, pages 387–396. Springer, 2011.

[12] S. Vakulenko, L. Nixon, and M. Lupu. Character-based neural embeddings for tweet clustering. *arXiv preprint arXiv:1703.05123*, 2017.

[13] S. Vosoughi, P. Vijayaraghavan, and D. Roy. Tweet2vec: Learning tweet embeddings using character-level cnn-lstm encoder-decoder. In *Proceedings of the 39th International ACM SIGIR conference on Research and Development in Information Retrieval*, pages 1041–1044. ACM, 2016.

[14] W. Xu, A. Ritter, C. Callison-Burch, W. B. Dolan, and Y. Ji. Extracting lexically divergent paraphrases from twitter. *Transactions of the Association for Computational Linguistics*, 2:435–448, 2014.

[15] F. M. Zanzotto, M. Pennacchiotti, and K. Tsioutsiouliklis. Linguistic redundancy in twitter. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, pages 659–669. Association for Computational Linguistics, 2011.