Visualizing Wikipedia using t-SNE

Jasneet Singh Sabharwal
Simon Fraser University
8888 University Drive
Burnaby, BC
jsabharw@sfu.ca

Abstract

This paper presents an approach on visualizing high-dimensional data of Wikipedia. Visualizing data is an important and well studied problem. Visualizing high-dimensional data can viewed as a dimensionality reduction problem. Over the past few decades many techniques have been proposed for linear and non-linear dimensionality reduction. In this paper we use one such non-linear dimensionality reduction technique called t-Distributed Stochastic Neighbor Embedding and specifically the version using Barnes-Hut algorithm for approximation. We also show that using semantic role labels as features for dimensionality reduction, we are able to visualize well clustered data with clusters containing similar data.

1 Introduction

Thanks to advances in internet technologies, sensing technologies, data management, mobile phone penetration, etc., each day the world is producing digital data at an astonishing rate. As of 2012, almost 2.5 exabytes of data was being created each day and the numbers keep on increasing. Analysing and understanding the large amount of data is an important problem as, within this large amount of data lies a wealth of information valuable for businesses, governments and for people.

Over the course of centuries, humans have evolved to understand patterns and trends visually rather than by looking at raw text and numbers. Visualizing the data would help in replacing mind numbing calculations and searching by displaying the data in a visual format which is more understandable and would help in improving inferences, comprehension and decision making. Computer scientists, psychologists and statisticians have studied how different visualizing techniques help in understanding data containing numbers, categories and networks and (Heer et. al., 2010) mention that spatial positions (like in scatter plots, line and bar charts) help in easier understanding of numerical data.

Visualizing low-dimensional data (2 or 3 dimensions) is an easy task and various visualizing techniques can be employed to understand the data. But, visualizing high-dimensional data is an important and tough problem which has been studied very well over the past few decades. (Oliveira and Levkowitz, 2003) have reviewed various high-dimensional data visualizing techniques like iconographic (Chernoff faces, stick figure, drift weed, etc.), geometric (RadViz, GridViz, circular parallel coordinates, etc.), pixel-based (circle segments, space filling curves, etc.) and hierarchical (dimension stacking and worlds-within-worlds). But these techniques are not viable enough to work on real world high-dimensional data.

The problem of visualization can also be viewed as a dimensionality reduction problem in which the goal is to map high-dimensional data into lower-dimensions (preferably 2 or 3 dimensions). In this we try to preserve the structure of the data in low-dimensions as it was in high-dimensions. Principal Component Analysis (Hotelling, 1933) and Multidimensional Scaling (Torgerson, 1952) are the standard linear dimensionality reduction techniques which focus on keeping dissimilar data far apart in lower dimensions. On the other hand, techniques like Stochastic Neighbor Embedding (Hinton and Roweis, 2003), t-Distributed Stochastic Neighbor Embedding (t-SNE) (Maaten and Hinton, 2008), Locally Linear Embedding (Roweis and Saul, 2000), Curvilinear Component Analysis (Demartines and Herault, 1997) and Maximum Variance Unfolding (Weinberger et. al., 2004) are nonlinear dimensionality reduction tech-

niques that focus on keeping similar datapoints close together in lower dimensions.

(Maaten and Hinton, 2008) showed that most of the nonlinear dimensionality reduction techniques (except t-SNE) perform strongly on artificial data sets but their visualization of real, high-dimensional data sets is poor. Whereas, t-SNE outperforms the other techniques and captures most of the local structure while revealing global structure like presence of clusters at various scales. t-SNE uses Gaussian distribution for calculating the probability of data points in higher dimension and uses Student-t distribution for representing points in lower dimension. It maps the points in higher dimension into lower dimension by minimizing the Kullback-Leibler divergence and for that it uses gradient descent technique. Due to this process, t-SNE runs in $O(N^2)$ time, which is fine for data sets containing few thousand records but, becomes slow when expanding to data sets containing several thousand to million records.

(Maaten, 2013) proposed an approach to perform t-SNE in $O(N \log N)$ time. They proposed an approach in which they use vantage-point trees to compute sparse pairwise similarities between input data records, and they used a variation of Barnes-Hut algorithm to approximate the forces between the corresponding records in the low-dimension embedding. The idea behind this approach is that multiple records (close to each other) which are far from the current record in high-dimensional space would exert less forces on the current record and thus instead of calculating the force for each far record, they approximate the force being exerted by only calculating for one far record and multiplying the force by the number of records within close vicinity of the far record.

2 Approach

In our work, we try to analyse the data in Wikipedia. Wikipedia is a collaboratively edited free encyclopedia which contains approximately 4.4 million records. Visualizing the articles of Wikipedia would help us in identifying various patterns, example, what kind of articles are similar i.e. is there any similarity between articles on war and articles on economy. For this work we utilized article summaries from Lensing Wikipedia[1] dataset. The dataset contains events related to human history and the important fields present in every record are *event, arg0, arg1, roleArg0, roleArg1, latitude, longitude, title, year, description, tokenized description* where *description* is a snippet of text from the Wikipedia article, *latitude* & *longitude* denote the location where the event occurred, *event* denotes the type of event, *arg0*, *arg1*, *roleArg0* & *roleArg1* denote the outputs of semantic role labelling.

As we want to visualize Wikipedia articles, we had to convert each article into a feature vector and we will explain more about the chosen features in the Experiment section. The feature vectors generated were in a very high dimensional space, therefore, we first had to bring down the dimensions of the data to 2 dimensions. For this, we utilized the Barnes-Hut-SNE as proposed by (Maaten, 2013). The reason for choosing Barnes-Hut version of t-SNE was because it runs in $O(N \log N)$ time as compared to the standard t-SNE which runs in $O(N^2)$ time. Since we will be running it on large number of high-dimensional records, that is why the time complexity was very important. Even though in the Barnes-Hut version, the forces exerted by data points on each other is approximated, the visualizations produced as compared to standard t-SNE are very similar as is the nearest neighbor error.

After performing dimensionality reduction, we had to visualize the data. For this, we utilized the power of D3[2], a JavaScript library for generating powerful visualizations. We visualize our data by generating a scatter plot in 2-dimension Cartesian Coordinate System. The tool also has the zoom functionality which not only allows us to study the global structure shown by the visualization but also allows us to see in detail the small clusters in detail by zooming in on them.

3 Experiment and Results

In our goal of analysing and understanding data in Wikipedia using Barnes-Hut version of t-SNE, we experimented with the features generated for performing t-SNE. In the following sections, we explain the 2 kinds of features that we used to visualize our data.

[1] Lensing Wikpedia: `http://lensingwikipedia.cs.sfu.ca`

[2] Data Driven Documents (D3): `http://d3js.org/`

3.1 Bag of Words

We created our corpus by extracting text from the *description* section of the dataset and all stop words were removed. We then created a dictionary of all words present in our corpus. For each record we then checked, which of the words in the dictionary were present in the record and what was the count of each present word (term frequency). In our test, each feature vector had 30,708 dimensions. Due to the very high-dimensionality of the data, we were only able to test for 15,000 records.

3.2 Semantic Role Labels

For this experiment, we created our corpus by extracting the semantic role labels and using only those labels as the features. So, for each document we check which role labels are present from our dictionary of role labels. This reduces the dimensionality of our feature vectors to 1834 dimensions. Due to this, we were able to visualize our complete data set of 39,170 records. For these features, we also tested for perplexity value of 20, 30 and 40.

3.3 Result

Figure 1 shows the visualization of the data with bag of word features. We also had set the parameter *perplexity* to the default value of 30. Figure 2 shows the visualization for the same 15,000 records as in Figure 1 with the default perplexity value of 30 but with semantic role label features. Comparing the two images, we can see that by using the semantic role label features, we get better clusters of data.

Figure 3, Figure 4 and Figure 5 show the visualization of the complete 39,170 records with *semantic role label* features with perplexity values of 20, 30 and 40 respectively. We can easily observe that perplexity value of 40 gives us the best distinct clusters. To understand how well the clusters were being formed, we digged deeper into our visualization with perplexity = 40 and we observed that the clusters being formed were meaningful and contained similar data. Figure 6 shows the cluster containing records with *event* tags of *defeated, defeat and defeats* and even the text of these records showed that the clusters were indeed of data related to defeats. We also studied more clusters, Figure 7 shows records with event tags of *established, found and founded* and similarly Figure 8 shows records related to killing and executed.

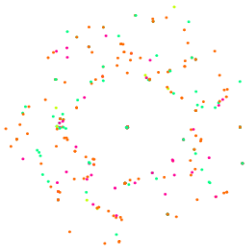

Figure 6: Cluster containing data with event tag of defeated, defeat and defeats generated by semantic role label features on complete dataset and perplexity = 40. Colours are based on *event* tag in our dataset

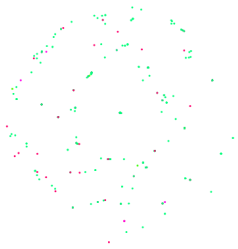

Figure 7: Cluster containing data with event tag of established, founded and found generated by semantic role label features on complete dataset and perplexity = 40. Colours are based on *event* tag in our dataset

4 Conclusion

This paper presented an approach on how we can utilize the Barnes-Hut version of t-Distributed Stochastic Neighbor Embedding to visualize high dimensional data of Wikipedia. Using this approach we were able to visualize well clustered data by utilizing semantic role label features.

Acknowledgments

We would like to thank Maryam Siahbani for providing the Lensing Wikipedia data set.

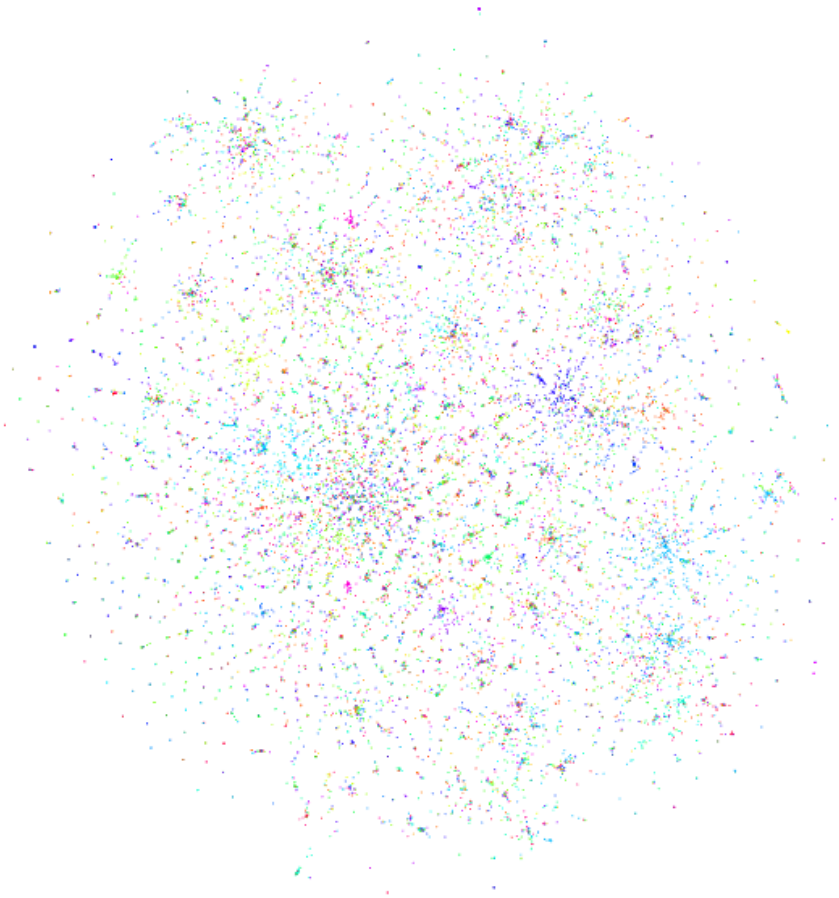

Figure 1: Bag of word features for 15,000 records with perplexity = 30. Colours are based on *event* tag in our dataset

Figure 8: Cluster containing data with event tag of kill, killed, kills and executed generated by semantic role label features on complete dataset and perplexity = 40. Colours are based on *event* tag in our dataset

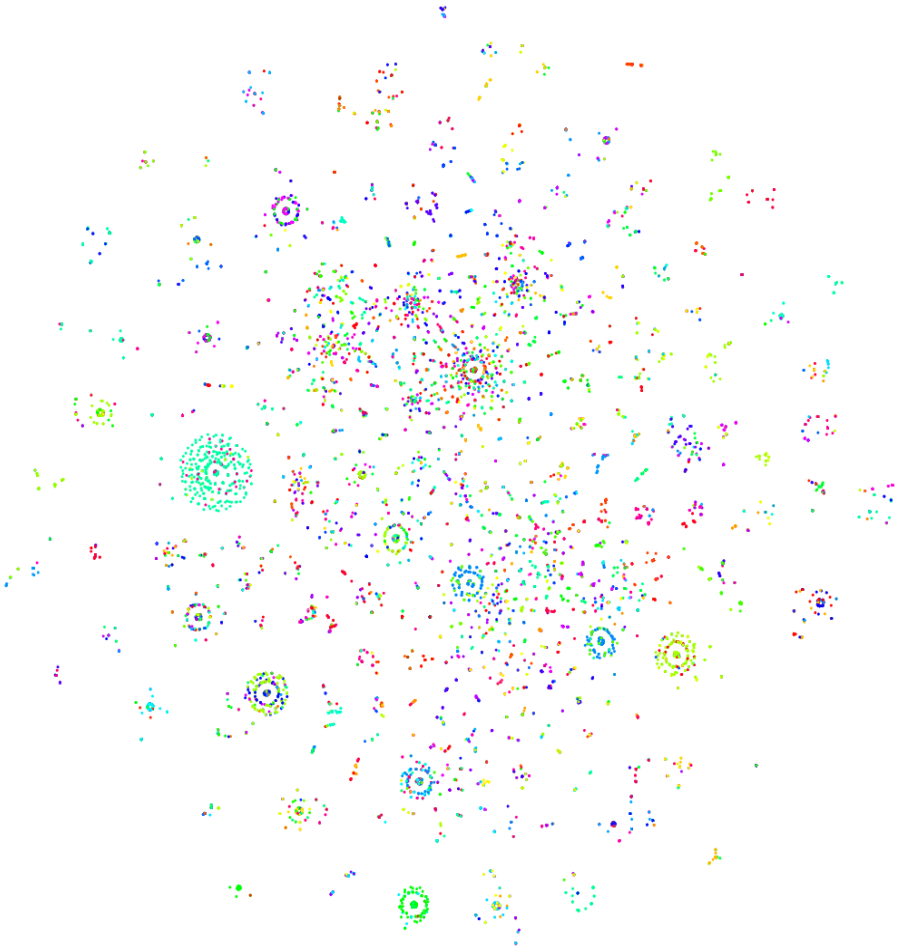

Figure 2: Semantic role label features for 15,000 records with perplexity = 30. Colours are based on *event* tag in our dataset

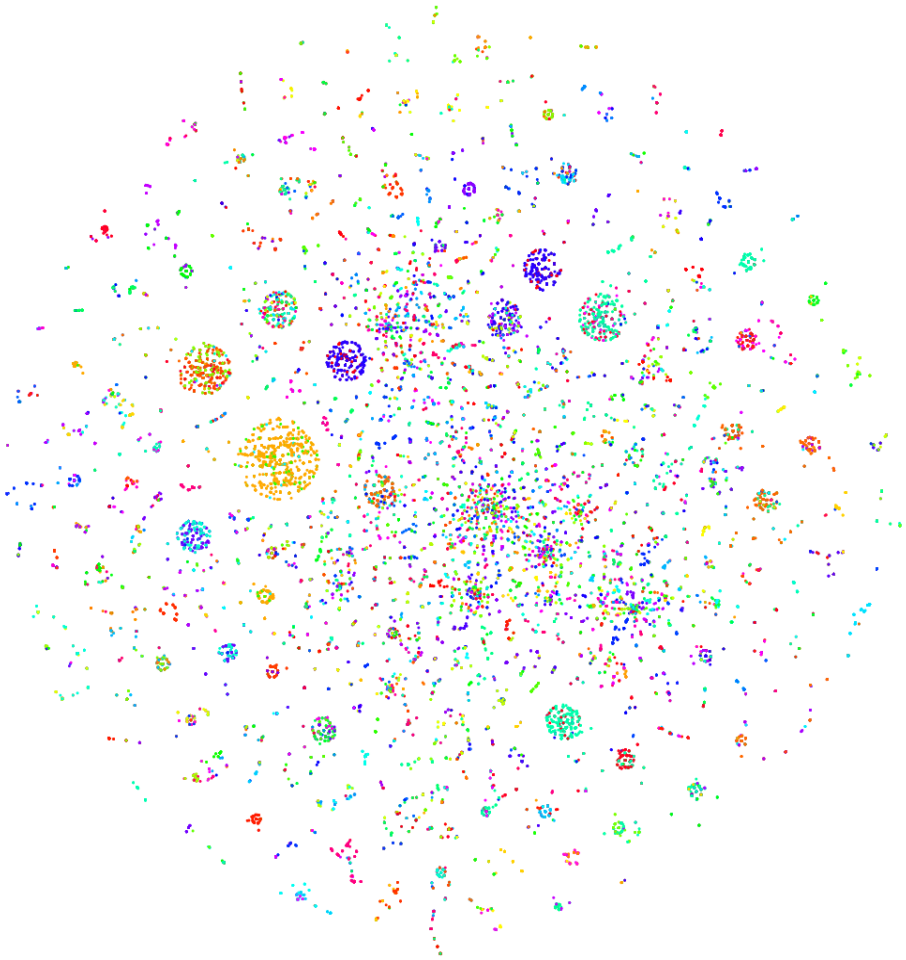

Figure 3: Semantic role label features for all records with perplexity = 20. Colours are based on *event* tag in our dataset

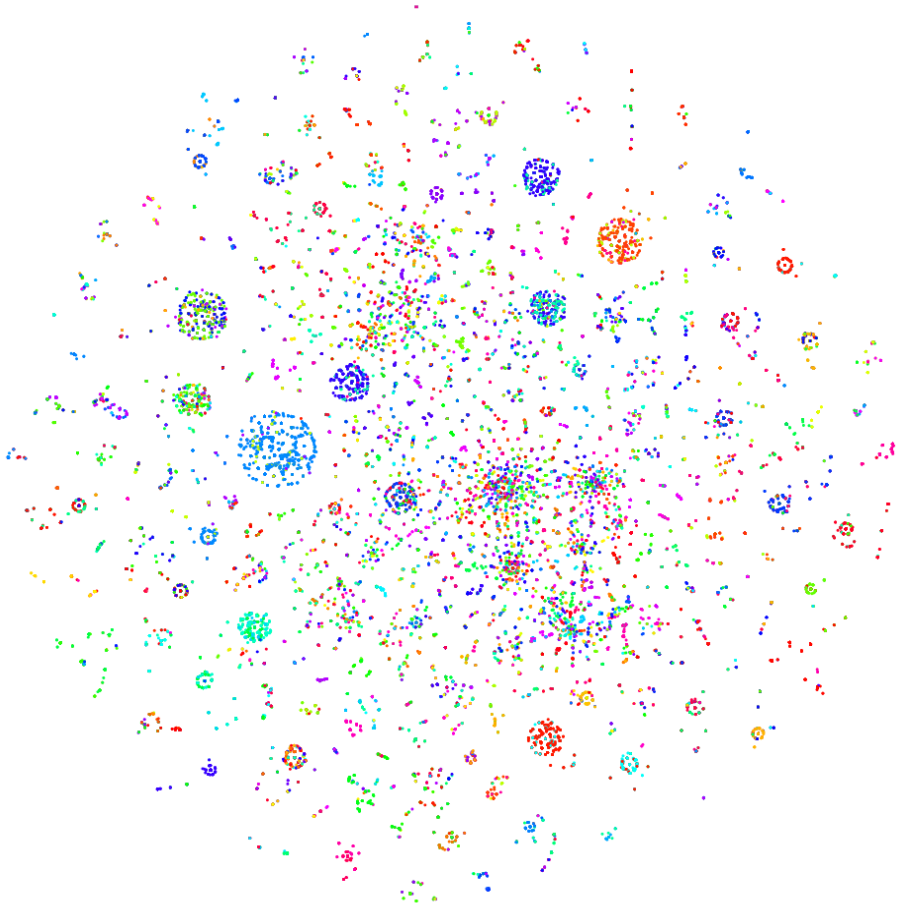

Figure 4: Semantic role label features for all records with perplexity = 30. Colours are based on *event* tag in our dataset

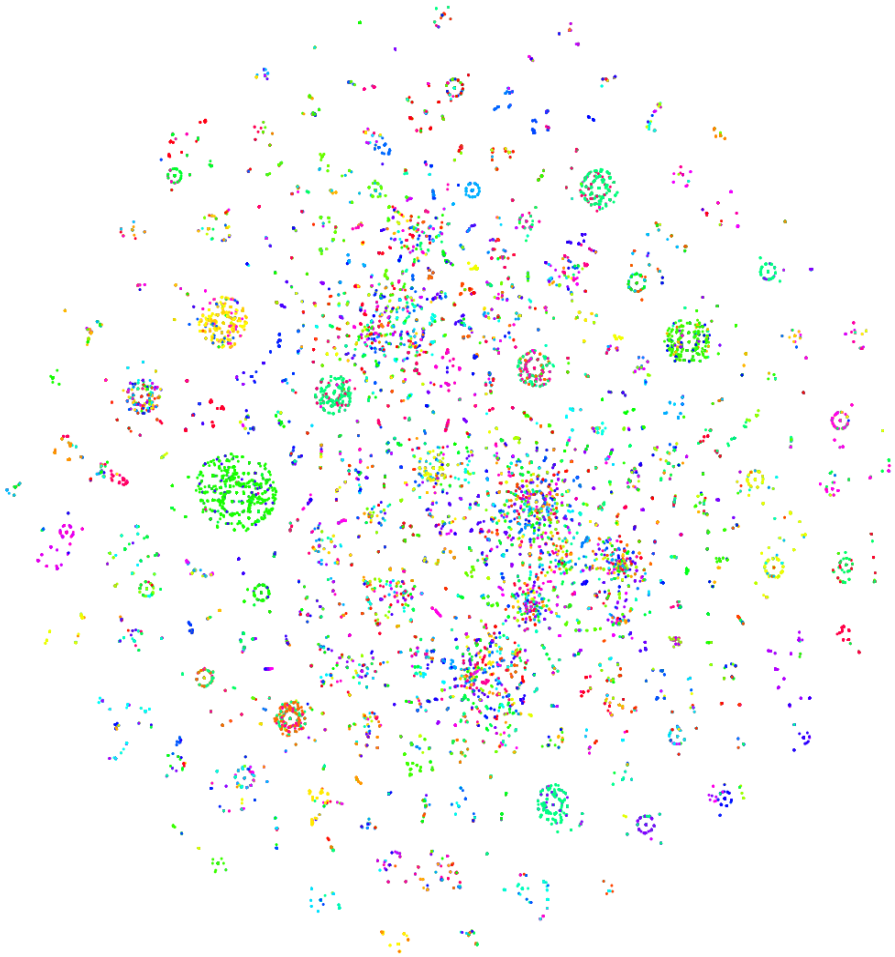

Figure 5: Semantic role label features for all records with perplexity = 40. Colours are based on *event* tag in our dataset

References

Jeffrey Heer, Michael Bostock, Vadim Ogievetsky 2010. *A tour through the visualization zoo.*. Association for Computing Machinery.

M.C.F de Oliveira, H. Levkowitz 2003. *From visual data exploration to visual data mining: a survey*. IEEE Transactions on Visualization and Computer Graphics.

H. Hotelling 1933. *Analysis of a complex of statistical variables into principal components*. Journal of Educational Psychology.

Warren S. Torgerson 1952. *Multidimensional scaling: I. Theory and method*. Psychometrika.

Geoffrey E. Hinton, Sam T. Roweis 2003. *Stochastic neighbor embedding*. Advances in Neural Information Processing Systems 15.

L.J.P. van der Maaten, Geoffrey E. Hinton 2008. *Visualizing High-Dimensional Data Using t-SNE*. Journal of Machine Learning Research 9.

Sam T. Roweis, Lawrence K. Saul 2000. *Nonlinear Dimensionality Reduction by Locally Linear Embedding*. Science.

P. Demartines, J. Herault 1997. *Curvilinear component analysis: a self-organizing neural network for nonlinear mapping of data sets*. IEEE Transactions on Neural Networks.

Kilian Q. Weinberger, Fei Sha, Lawrence K. Saul 2004. *Learning a Kernel Matrix for Nonlinear Dimensionality Reduction*. Proceedings of the Twenty-first International Conference on Machine Learning.

L.J.P. van der Maaten 2013. *Barnes-Hut-SNE*. Proceedings of the International Conference on Learning Representations.