CSCI4/6360 Final Report for Victorian Literature

Zirak, Aleksandr, Jialin
Department of Computer Science
University of Georgia
Athens, GA 30602

December 6, 2019

Abstract

We propose a novel approach for the classification of words into predefined categories that requires only one seed for each desired categories and can be used on a large amount of data. Our data pipeline relies on the pretrained GloVe embedding and the semi-supervised MultiRank Walk algorithms which excels at classification of large datasets with little data points.

This approach can then be applied to texts written by Victorian-era authors to determine which themes appeared most frequently in their writings, and get the proportions for four themes: Class, Religion, Industrialization, and Science.

1 Introduction/Background Information

During the Victorian Period (June 1837 to January 1901), many different aspects changed in England including religion, industry, science, class. Since popular entertainment, such as literature, is often a reflection of the sentiments of the time period of publication, by documenting the themes in Victorian literature during this time we can see the effect social, economic, and political change has on society. This will help us to better contextualize events of the past, as well as prepare us for the effect future societal, economic, and/or political change will have on the resulting literature.

Our purpose is to analyze texts written by four English authors during the Victorian period. Specifically, we analyzed 161 novels and poems written by Bronte Charlotte, Dickens Charles, Hardy Thomas and Trollope Anthony (data is pulled from the GitHub repository of Project Gutenberg at `https://github.com/DigiUGA/Gutenberg_Text`). We are mainly interested in the following questions:

(a) Does victorian literature focus primarily on class, science and religion, and industrialization?

(b) Determine the common themes by considering the frequency of words in texts.

(c) Analyze texts in the victorian genre by English authors and compare their themes.

2 Method

The steps we have taken is in figure 1. After deleting the words we don't want, we give all words in the corpus a 300 dimension coordinate (by Glove) to represent the word's location in the vocabulary space. And then, by Multi-Rank Walk, each word was classified into one of the four themes. Finally, the proportion of themes in each text can be calculated.

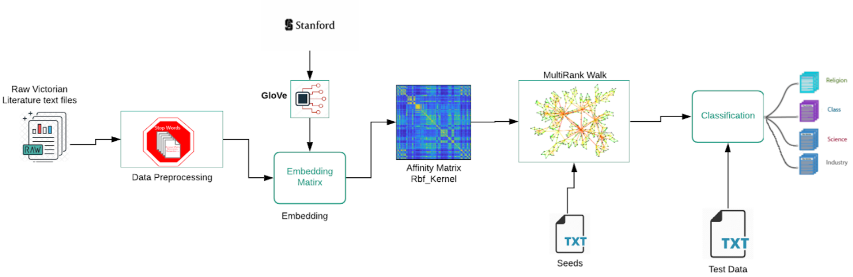
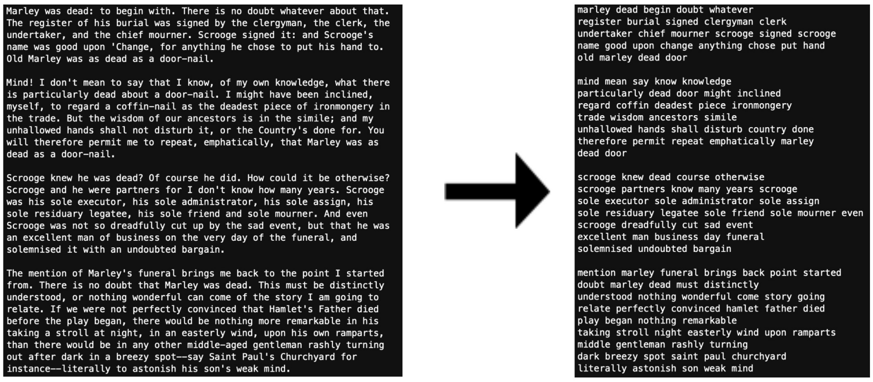


Figure 1: The steps of classifying texts

2.1 Removing undesired words by NLTK

The texts of the four authors listed above that were pulled from the Project Gutenberg repository presented themselves in a raw form. Since literary themes are not influenced by prepositions, numbers, capitalization, etc. The text corpus had to be preprocessed and cleaned. In order to keep words that have bearing on literary themes and could be analyzed, we passed the text through several filters created with NLTK. These filters include tokenizing all the words, removing punctuation, conversion to lowercase, removing stop words pulled from two libraries (approximately 1050 words), and removing all numbers. Figure 2 is an example shows what will happen to four paragraphs after using NLTK filters. We tried to apply stemming which would find the root of a word, but felt that it did not sufficiently consolidate the total vocabulary size. The words were finally grouped in a set which constricted it to only unique words, of which there were approximately 6,470,035 words (with repetition)1 are kept in these 161 novels and poems (Table 1). Figure 2 is an example to show what will happen after we use NLTK.



```
Marley was dead: to begin with. There is no doubt whatever about that.
The register of his burial was signed by the clergyman, the clerk, the
undertaker, and the chief mourner. Scrooge signed it: and Scrooge's
name was good upon 'Change, for anything he chose to put his hand to.
Old Marley was as dead as a door-nail.

Mind! I don't mean to say that I know, of my own knowledge, what there
is particularly dead about a door-nail. I might have been inclined,
myself, to regard a coffin-nail as the deadest piece of ironmongery in
the trade. But the wisdom of our ancestors is in the simile; and my
unhallowed hands shall not disturb it, or the Country's done for. You
will therefore permit me to repeat, emphatically, that Marley was as
dead as a door-nail.

Scrooge knew he was dead? Of course he did. How could it be otherwise?
Scrooge and he were partners for I don't know how many years. Scrooge
was his sole executor, his sole administrator, his sole assign, his
sole residuary legatee, his sole friend and sole mourner. And even
Scrooge was not so dreadfully cut up by the sad event, but that he was
an excellent man of business on the very day of the funeral, and
solemnised it with an undoubted bargain.

The mention of Marley's funeral brings me back to the point I started
from. There is no doubt that Marley was dead. This must be distinctly
understood, or nothing wonderful can come of the story I am going to
relate. If we were not perfectly convinced that Hamlet's Father died
before the play began, there would be nothing more remarkable in his
taking a stroll at night, in an easterly wind, upon his own ramparts,
than there would be in any other middle-aged gentleman rashly turning
out after dark in a breezy spot--say Saint Paul's Churchyard for
instance--literally to astonish his son's weak mind.
```

```
marley dead begin doubt whatever
register burial signed clergyman clerk
undertaker chief mourner scrooge signed scrooge
name good upon change anything chose put hand
old marley dead door

mind mean say know knowledge
particularly dead door might inclined
regard coffin deadest piece ironmongery
trade wisdom ancestors simile
unhallowed hands shall disturb country done
therefore permit repeat emphatically marley
dead door

scrooge knew dead course otherwise
scrooge partners know many years scrooge
sole executor sole administrator sole assign
sole residuary legatee sole friend sole mourner even
scrooge dreadfully cut sad event
excellent man business day funeral
solemnised undoubted bargain

mention marley funeral brings back point started
doubt marley dead must distinctly
understood nothing wonderful come story going
relate perfectly convinced hamlet father died
play began nothing remarkable
taking stroll night easterly wind upon ramparts
middle gentleman rashly turning
dark breezy spot saint paul churchyard
literally astonish son weak mind
```

Figure 2: An example: paragraphs after NLTK

Table 1: Data Summary for all texts

Authors	Novels	Words before NLTK	Words w & w/o Repetition
Hardy, Thomas	26	2472800	1042777/ 40254
Bronte, Charlotte	5	540285	248647/20648
Dickens, Charles	84	8555461	3929794/54666
Trollope, Anthony	46	2941267	1248817/33495

2.2 Embeddings by GloVe[1]

Once the data is cleaned, we want to establish a relationship between words such that the similarity between words can be measured. The challenge comes that how to represent discrete words by some continuous features, or how to give each word a vector representation to show its location in the "vocabulary space". The solution is the Global Vectors for Words Representation (GloVE)[2] which represents millions of words in high dimensional Euclidean space. This process needs tons of trainings on massive web datasets, and was done by Jeffrey Pennington, Richard Socher, and Christopher D. Manning at Stanford University, which is called Global Vectors (GloVe). This embedding was trained on 840 billion tokens and mapped approximately 2.2 million unique words to 300 dimensions, i.e, it represents all 2.2 millions words with different vectors of length 300.

For every single word in our vocabulary set, we find the resulting embedding vector and stacked all the vectors for each word into a matrix while also noting which word appeared on which row. Since GloVe is trained using modern vocabulary and the usage of words is different in Victorian period from nowadays, not all words of the authors' texts received an embedding. The final matrix that was generated was 56,215 by 300 where the rows were unique words, and the columns where the actual embeddings of each word.

2.3 Generating Affinity Matrix

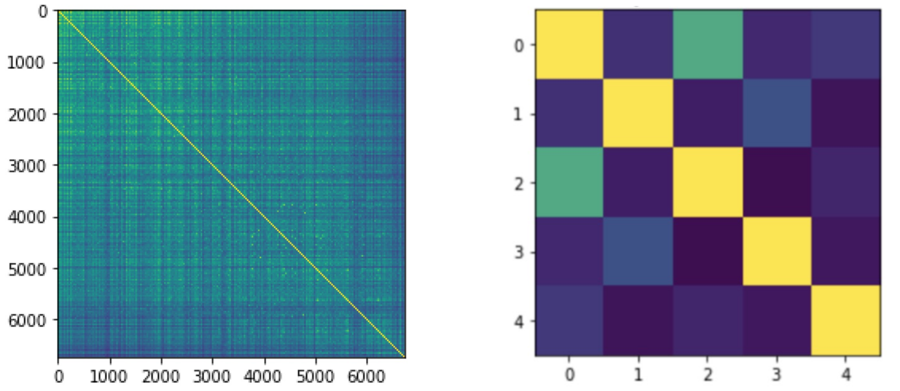
The previous step generates a matrix of all relevant vocabulary words that appear in our dataset and their embeddings, or representations in space. The Euclidean distance can be taken between each of these words to determine their similarities, but we want to calculate the difference between every word and every other word and to constrain the difference between 0 and 1.

As a result, the embedding matrix is run through the radial basis function kernel [2] that generates an affinity matrix. Since the embeddings are in Euclidean space, the difference between word A and word B is the same as the difference between word B and word A, the resulting graph after the kernel function is symmetric which will simplify further calculations. The resulting affinity matrix is 56,215 by 56,215, which is plotted in Figure 3(a). Here, in each tiny cell, yellow color means the meaning of these pair of words have close meaning, while dark blue color means the difference of these two words are quite large.

[1] `https://nlp.stanford.edu/projects/glove/`

[2] Radial Basis Function kernel.

We also create an example to help understand affinity matrix. Figure 3(b) is the affinity matrix for five words: "man", "computer", "women", "science", and "frog". From this figure, the 2-th row 0-th column is green, which means our embedding in part 2.2 could assign "man" and "women" with two close coordinates; for "computer" and "science" (which is on 3-th row, 2-th column), our embedding also assign them two close vector representations. With this affinity matrix, we could classify every words into one of the theme, which will be explained later.



(a) for the whole text data

(b) man, computer, women, science, frog

Figure 3: Affinity Matrix Visualization

2.4 MultiRank Walk

One the relationship between words has been established and converted to an affinity matrix, it is possible to begin clustering the words into different predefined categories. In order to do this, we propose using the semi-supervised MultiRank Walk algorithm created by Lin and Cohen [1]. The algorithm excels at classifying large volumes of data with minimal amount of labelled data points. Specifically, MRW takes a graph $G = (V, E)$, where the nodes in G are instances X composed of unlabelled instances X^U and labelled instances X^L with corresponding labels Y^L and finds labels Y^U for all unlabelled data points X^U.

The main equation behind MRW is given by $\vec{r} = (1-d)\vec{u} + dW\vec{r}$, where d is a predefined damping factor that controls "teleportation" across the graph, $\vec{u}$ is the normalized teleportation vector and W_{ij} is the weighted transition matrix of graph G from vertex i to vertex$j$.

To begin MRW, we input the input graph $G = (V, E)$ which is the precalculated affinity matrix. We first find the matrix D which calculates the sum of the rows in the affinity matrix and puts them on the diagonal such that the shape of is the same as the shape of the affinity matrix. Next, the weighted transition matrix W can be calculated. W has the same shape as and the affinity matrix and is calculated by $W_{ij} = \frac{A_{ij}}{D_{ii}}$.

We can then create the seeding vectors $\vec{u}$ of which there are four (one for each class). The seeding strategy will be discussed in a later section, but for each class we had to provide word(s) that would correspond to the literary theme. For example, "coal", "iron", "steel" would be in the Industrialization category and "bible", "God", "church" would be in the Religion category. For each word ("seed") in a category, we would note which index they appear in in the affinity matrix. Then those indices in the $\vec{u}$ vector would be set to a one, with zeroes everywhere else. The process would then be repeated for all categories such that we have four seeded $\vec{u}$ vectors. The vectors were then normalized with the l_1 norm such that the sum of elements in each vector was 1.

Once all the pieces are calculated, we iteratively apply the formula above to find four ranking vectors $\vec{r}$ (one for each class). We settled on using a damping factor d of 0.95 and iterated over the formula for 100 epochs. This results in a series of $\vec{r}$ vectors that can be used to classify words into their categories.

For each word that we classify, we get the word's unique number and check that index in all four $\vec{r}$ vectors. Whichever $\vec{r}$ vector had the highest value at that index would be the category that we placed the word into. We can then simply query the $\vec{r}$ vectors each time we want to classify a word.

2.5 Seeding

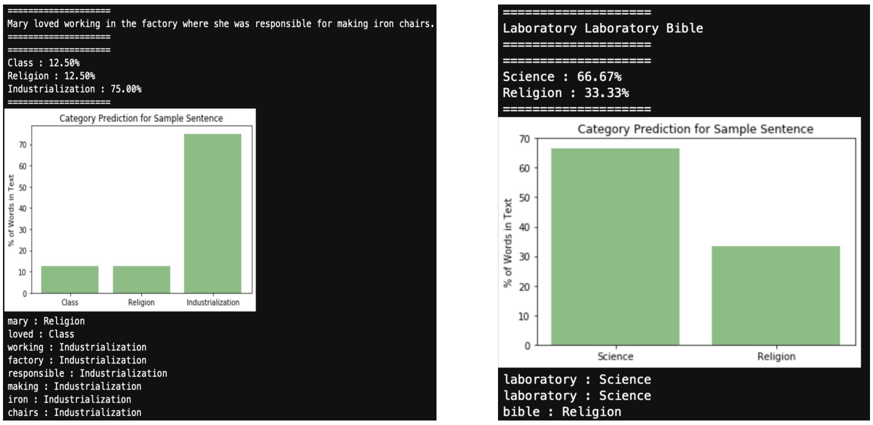
We evaluated multiple different methods of seeding the $\vec{u}$ vectors above. The seeding works by giving MRW a data point that has been categorized into the correct class. Our original approach had us hand labelling as many words as possible into the four categories then using those as seeds. However, we discovered that whichever category contained the most seeds would then dominate the classes that would be predicted.

We considered labelling roughy one-hundred words that would be evenly split across the classes, but realized that we could provide only one word for each class ("religion" in the religion category, "industry" in the industrialization category, etc.) and let MRW more accurately decide categories. This approach resulted in less domination of predictions by a single category and offered nice variance between predicted classes that reflected the actual themes in the novels.

3 Evaluation

The text corpus was passed through the same NLTK filters to generate a cleaned data set, and Multi-rank Walk predict the theme for all words. Figure 4 illustrates how our method works. In Figure 4(a), the input is a sentence "Mary loved working in the factory where she was responsible for making iron chairs" and after the NLTK filters, the words "in", "the", "where", "she", "was" and "for" were deleted and all words were converted to lowercase. Then, we use Multi-Rank Walk predict the theme for each word. "Mary" was classified into "Religion"; "loved" was classified into "Class"; "working", "factory", "responsible", "making", "iron" and "chairs" are all classified into "Industrialization". Hence, we get 12.5% Class, 12.5% Religion and 75% Industrialization.

For evaluation, duplicate words are not removed as repeated words have a strong effect on the theme of the text. In Figure 4(b), the data was partitioned into separate portions corresponding to each individual novel and each cleaned word for that novel was categorized using MRW and the $\vec{r}$ vectors. If certain words appear more frequently than others, it will influence the results more.



```
====================
Mary loved working in the factory where she was responsible for making iron chairs.
====================
====================
Class : 12.50%
Religion : 12.50%
Industrialization : 75.00%
====================
```

```
mary : Religion
loved : Class
working : Industrialization
factory : Industrialization
responsible : Industrialization
making : Industrialization
iron : Industrialization
chairs : Industrialization
```

(a) A whole sentence

```
====================
Laboratory Laboratory Bible
====================
====================
Science : 66.67%
Religion : 33.33%
====================
```

```
laboratory : Science
laboratory : Science
bible : Religion
```

(b) Testing repititions

Figure 4: Method results evaluation

For each novel, the predicted category is pooled for each word which results in a prediction for the theme of the novel. The individual percentages of the categories can be calculated as well.

Finally, we passed on our result to the students in the English department and they managed to identify themes either in specific novels, or for authors that they knew well.

4 Results

4.1 Prediction for each novel/poem

We picked three examples to visualize our prediction in Figure 5. The bar in figures represents the proportion of each themes for the specific novel/poem. From this figure, *The Book of the Homeless* has more `science` words than others; *A Group of Noble Dames* has more `class` words than others; *Neither Dorking nor the Abbey* has more `industrialization` words than others. [3]

4.2 Prediction for overall Victorian Literature

Based on Figure 6, the results show that Victorian-era novels focused primarily on class (34.06%) and industrialization (31.31%) with the two categories being

[3] All these novels are by Hardy Thomas, but with different themes.

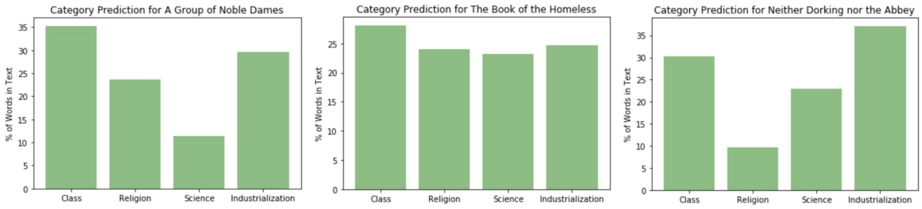


Figure 5: Prediction for each novel/poem

roughly equivalent in presence, though class typically has a higher percentage in novels. Religion (22.08%) is a category that typically more present than science, but less so that class and industrialization. Finally, science (12.64%) was the least present category but did occasionally have high presence.

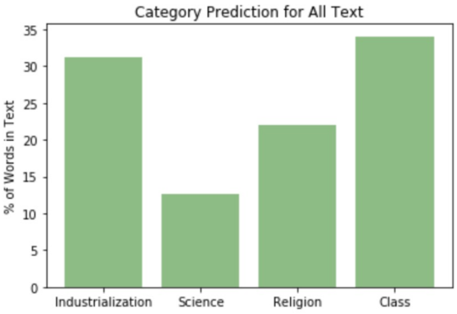


Figure 6: Prediction for overall Victorian Literature by these 4 authors

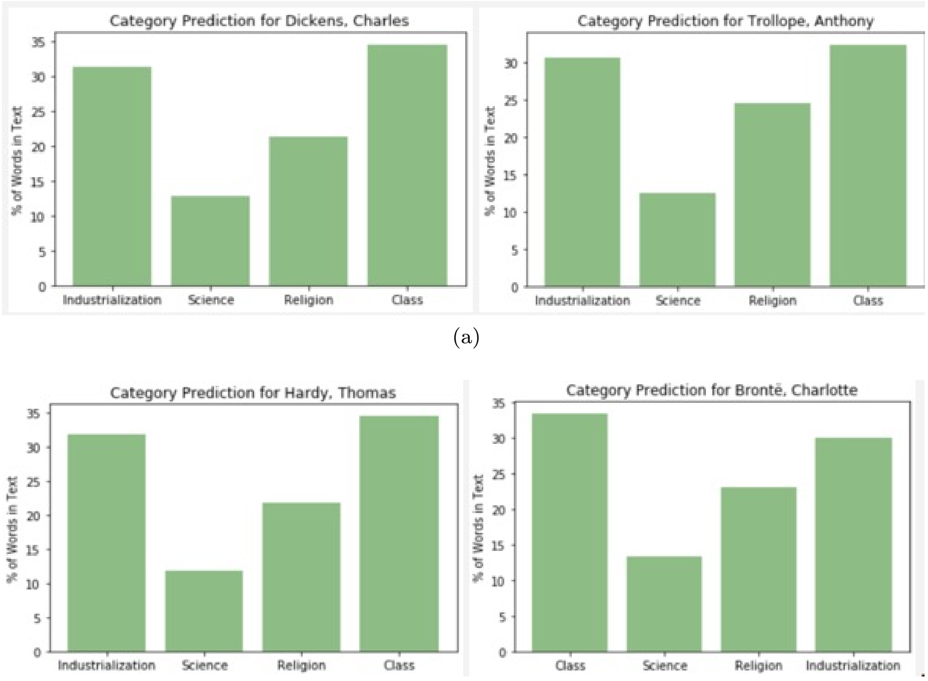
4.3 Comparison among these four authors

From Figure 7, there is very little difference in the themes captured by these authors. All of them predominantly wrote about class and industrialization, though their novels do capture a variance in themes as seen in Figure 5. Our english collaborators looked at our results and confirmed that class should be a major category as poverty and social constraint was an omnipresent issue. Additionally they thought that religion would be a frequent theme and our results show that that is the case.

We also put the numerical proportion of four themes in Table 2.

Table 2: Theme proportions for 4 authors

Authors	Industrialization	Science	Religion	Class
Hardy, Thomas	31.83	11.86	21.77	34.54
Bronte, Charlotte	30.10	13.35	23.06	33.49
Dickens, Charles	31.31	12.86	21.32	34.51
Trollope, Anthony	30.60	12.48	24.56	32.36



(a)

(b)

Figure 7: Comparison for 4 authors

Acknowledgments

We would like to express our sincere gratitude to our cooperators Taylor Biggar, Leila Mohammadizadeh and Eden Esemuede in the English course for coming up with these research questions, helping us on understanding the background of Victorian Literature, providing all raw data, sorting out further questions and comments.

Also we would like to thank Dr. Shannon Quinn's teaching and suggestions throughout this semester, and thank all the classmates on CSCI6360 for giving wonderful presentations.

References

[1] F. Lin and W. W. Cohen. Semi-supervised classification of network data using very few labels. In *2010 International Conference on Advances in Social Networks Analysis and Mining*, pages 192–199, 2010.

[2] 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.