8-2018

Theatrical Genre Prediction Using Social Network Metrics

Manisha Shukla
University of Arkansas, Fayetteville

Theatrical Genre Prediction Using Social Network Metrics

A thesis submitted in partial fulfillment
of the requirements for the degree of
Master of Science in Computer Science

by

Manisha Shukla
Rajasthan Technical University
Bachelor of Technology in Computer Science, 2010

August 2018
University of Arkansas

This dissertation is approved for recommendation to the Graduate Council.

Susan Gauch, PhD
Dissertation Director

Merwin Gordon Beavers, PhD
Committee Member

Michael S. Gashler, PhD
Committee Member

ABSTRACT

With the emergence of digitization, large text corpora are now available online that provide humanities scholars an opportunity to perform literary analysis leveraging the use of computational techniques. This work is focused on applying network theory concepts in the field of literature to explore correlations between the mathematical properties of the social networks of plays and the plays' dramatic genre, specifically how well social network metrics can identify genre without taking vocabulary into consideration. Almost no work has been done to study the ability of mathematical properties of network graphs to predict literary features. We generated character interaction networks of 36 Shakespeare plays and tried to differentiate plays based on social network features captured by the character network of each play. We were able to successfully predict the genre of Shakespeare's plays with the help of social network metrics and hence establish that differences of dramatic genre are successfully captured by the local and global social network metrics of the plays. Since the technique is highly extensible, future work can be extended for fast and detailed literary analysis of larger groups of plays, including plays written in different languages as well as plays written by different authors.

ACKNOWLEDGEMENTS

I would like to thank Dr. Susan Gauch for providing me with the opportunity to work with her during this project and her guidance throughout my graduate education. I also thank my thesis committee, Dr. Merwin Beavers and Dr. Michael Gashler, for reviewing my thesis.

I would also like to thank my husband, Mr. Pawan Mishra, for his support and encouragement throughout my studies at the University of Arkansas as well as for his understanding while I spent a majority of my time on this project.

TABLE OF CONTENTS

LIST OF PUBLISHED/ACCEPTED PAPERS

1. Chapter 1.2.2, Chapter 2 - Shukla, Manisha, Susan Gauch and Lawrence Evalyn. 2018. Theatrical Genre Prediction Using Social Network Metrics. In 10th International Conference on Knowledge Discovery and Information Retrieval, Seville, Spain. (Accepted).

1. INTRODUCTION

1.1 Background

In literary studies, the three key areas of research could be defined as philology (the study of words), bibliography (the study of books as objects), and criticism (the evaluation or interpretation of literary meaning). Particularly since the advent of New Criticism, “the basic task of literary scholarship has been close reading of texts” (Moretti 2011), which builds textual interpretations from precise study of specific words. Computational approaches to literature offer an alternate methodology for the work of literary study without close reading. “Distant reading” (Moretti 2011) takes many forms, including statistical topic models (Jockers and Mimno 2013), character profiling (Flekova and Gurevych 2015), character frequency analysis (Sack 2011), and sentiment analysis (Elsner 2015), as mentioned in Grayson et al. 2017. For computational methods to produce new literary insights, they must provide information about literary texts which is not easily accessible by reading them and must do so for more texts than it is feasible for a person to read. Our paper presents a distant reading method which may aid in the task of literary criticism using network graph analysis on social networks generated from the scripts of plays.

Here, we study the social networks of Shakespeare’s plays to establish a correlation between social network metrics and genre identification. Using character networks of Shakespeare’s plays we found that combinations of some of the global and local network metrics (Watts 2001) were able to distinguish plays belonging to different genres. This work has been used for literary analysis of the ambiguous genre of Shakespeare’s “problem plays” (Evalyn, Gauch, and Shukla, 2018).

1.2 Motivation

Social network analysis is well-established to study social groups. Some scholars have applied social network analysis to literary works e.g., plot analysis (Grayson et al. 2016), or for discovering character communities (Watts 2001), wherein nodes represent characters, and edges represent interaction between pairs of characters for plot analysis. However, because these graphs are handmade for a very small number of plays, almost no work has been done to study the ability of mathematical properties of network graphs to predict literary features. We address this gap by exploring correlations between the mathematical properties of networks and dramatic genre.

1.2.1 Why Social Networks of Plays?

Our work presents a distant reading method which may aid in the task of literary criticism using network graph analysis on social networks generated from the scripts of plays taking only characters and their interaction into consideration. It is focused on applying network theory concepts in the field of literature to explore correlations between the mathematical properties of the social networks of plays and the plays' dramatic genre, specifically how well social network metrics can identify genre. Almost no work has been done to study the ability of mathematical properties of network graphs to predict literary features. Since the technique is highly extensible, future work can be extended for fast and detailed literary analysis of larger groups of plays, including plays written in different languages as well as plays written by different authors.

1.2.2 Discussion

The relevance of graph density in distinguishing genres is visually obvious when individual comedy and history networks are compared. Histories feature highly dispersed networks, with large numbers of very minor characters, such as "First," "Second," and "Third" members of groups like soldiers and ambassadors (Figure 1). Comedies, in contrast, feature networks with far fewer characters, in which nearly everybody speaks to nearly everybody else at some point (Figure 2).

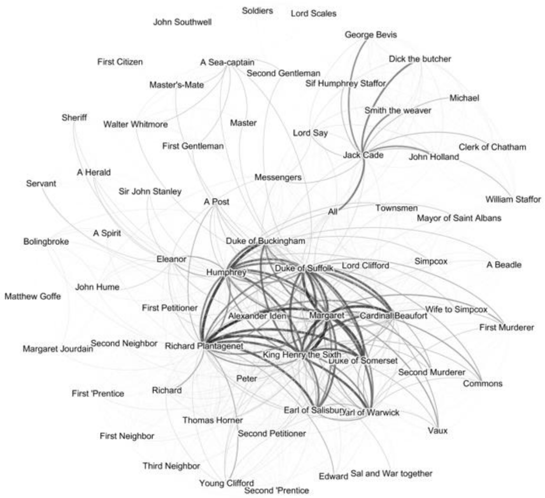


Figure 1: Network graph of The Second Part of King Henry The Sixth, a history.

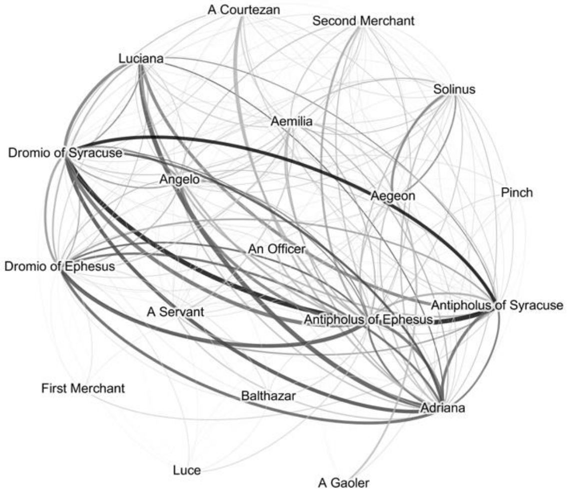


Figure 2: Network graph of The Comedy of Errors, a comedy.

Any single feature is insufficient, however, to fully distinguish the tragedies, which feature networks somewhere between history and comedy in their density and show more variety overall (Figures 3 and 4). Therefore, more complex metrics are needed in combination with each other to accurately identify all three genres.

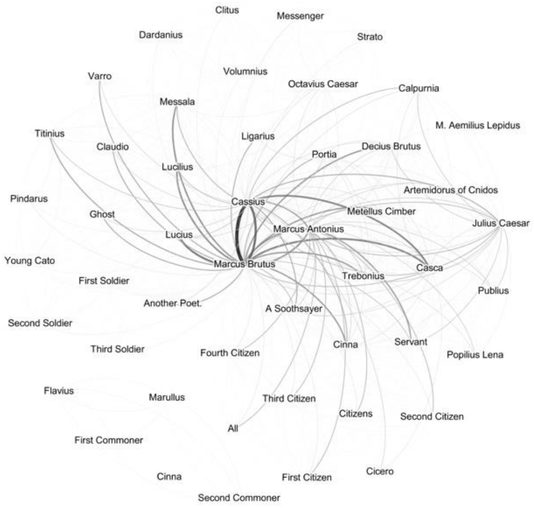


Figure 3: Network graph of Julius Caesar, a tragedy

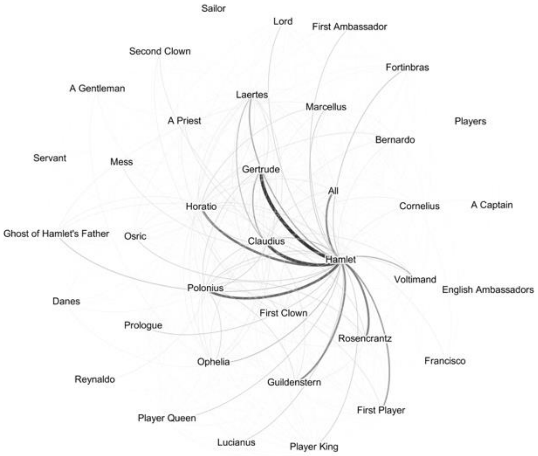


Figure 4: Network graph of Hamlet, a tragedy.

Our networks of the well-studied works of Shakespeare can provide a baseline against which to contextualize similar studies of other plays. The network graphs themselves provide a new insight into the plays, revealing the hidden shape of social relationships between characters.

The application of mathematical graph analysis to these networks provides a dramatically faster and more scalable way to determine important information about them, in this case their genre. The presented work is based on one central question: Can we develop a computational model that captures these differences and uses them for genre prediction?

1.3 Organization of this Thesis

In Chapter 2, we present a summary of related work on social networks in different fields and the literary world. Chapter 3 introduces a methodology for generating social networks of plays and presents which classifier, graph representation and metrics were chosen for classifying the plays by genre. In Chapter 4, we report on the different experiments that we conducted, their results and their evaluation. Finally, in Chapter 5, we present conclusions and discuss our ongoing and future work in this area.

1.4 References

1) Watts, D. 2001. "Small Worlds: The Dynamics of Networks between Order and Randomness", Princeton University Press.

2) Moretti, F. 2011. Network Theory, Plot Analysis. New Left Review, 68:80–102.

3) Flekova, and I. Gurevych. 2015. Personality Profiling of Fictional Characters using Sense-Level Links between Lexical Resources. In Proc. Conference on Empirical Methods in Natural Language Processing, pages 1805–1816.

4) Sack, G. 2011. Simulating plot: Towards a generative model of narrative structure. In 2011 AAAI Fall Symposium Series.

5) Evalyn, Lawrence, Susan Gauch, and Manisha Shukla. 2018. Analyzing Social Networks of XML Plays: Exploring Shakespeare's Genres. In Digital Humanities Conference 2018. https://dh2018.adho.org/en/analyzing-social-networks-of-xml-plays-exploring-shakespeares-genres/.

6) Elsner, M. 2015. Abstract Representations of Plot Struture. LiLT (Linguistic Issues in Language Technology), 12(5).

7) Jockers, M. L. and D. Mimno. 2013. Significant themes in 19th-century literature. In Poetics, 41(6):750–769.

8) Shukla, Manisha, Susan Gauch, and Lawrence Evalyn. 2018. Theatrical Genre Prediction Using Social Network Metrics. In 10th International Conference on Knowledge Discovery and Information Retrieval, Seville, Spain. (Accepted).

9) Grayson, Siobhán, Karen Wade, Gerardine Meaney, Jennie Rothwell, Maria Mulvany, and Greene Derek. 2016. Discovering structure in social networks of 19th century fiction. In Proceedings of the 8th ACM Conference on Web Science (WebSci '16). ACM, New York, NY, USA, 325-326.

2. RELATED WORK

This chapter presents related research on application of social networks in various fields and social network analysis in the field of literature.

2.1 Social Network Analysis

2.1.1 Social Networks

As Billah and Gauch observe, “Social network analysis (SNA) is not a formal theory, but rather a wide strategy for investigating social structures” (Billah and Gauch, 2015). These strategies borrow core concepts from sociometry, group dynamics, and graph theory (Watts 2001; Scott 2000; Wasserman and Faust 1994).

A social network graph is a set of vertices and edges (called a sociogram) where vertices represent social actors and edges represent social relations among the vertices. However, a social network is more than just a set of vertices and lines, as its structure contains implicit information about the social actors and their relationships. The graph representation of a social network offers a systematic and mathematical method for investigating these structures. Social network analysis is the process of investigating social network structures and ties through the use of network and graph theory concepts.

In social network analysis of human activities, the nodes can be connected by many kinds of ties, such as “shared values, visions, and ideas; social contacts; kinship; conflict; financial exchanges; trade; joint membership in organizations; and group participation in events, among numerous other aspects of human relationships” (Serrat 2017). However, regardless of the nature of the connection, “the defining feature of social network analysis is its focus on the structure of relationships” (Serrat 2017). The central assumption in SNA methodologies is that relationships between nodes are of central importance (Serrat 2017).

1.2.1 Current Research in Social Network Analysis

Social network analysis has been used in a wide variety of fields, with applications as diverse as disintegration models based on social network analysis of terrorist organizations (Anggraini et al. 2015), collaboration of scholars in graduate education (Chuan-yi, Xiao-hong, and Yi 2016), football team performance based on social network analysis of relationships between football players (Trequattrini, Lombardi, and Battista 2015), money laundering detection (Dreżewski, Sepielak, and Filipkowski 2015), and stress disorder symptoms and correlations in U.S. military veterans (Armour et al. 2017). In this paper, we explore application of social network in literary analysis, specifically in exploring how well social network metrics can identify genre without taking words into consideration which will lead it to potential possibilities of extension in future with variation in languages and authors.

2.2 Literary Analysis with Social Networks

Because dramatic performances enact social encounters, social network analysis translates surprisingly well to fictional societies. Stiller et al. have shown that social networks in Shakespeare's plays mirror those of real human interactions, particularly in size, clustering, and maximum degrees of separation (Stiller, Nettle, and Dunbar 2003).

Surveying the field of literary analysis using SNA, Moretti categorizes several types of analyses: "an empirical, quantitative and hierarchical description of literary characters (Jannidis et al. 2016), corpus-based analyses exploring options for historical periodization of literature (Trilcke et al. 2015) and types of aesthetic modelling of social formations in and by literary texts (Stiller, Nettle, and Dunbar 2003; Stiller and Hudson 2005; Trilcke et al. 2016)." Moretti himself uses social networks to examine the plots of three Shakespearean tragedies, and to contrast a few chapters in English and Chinese novels (Moretti 2011). Work following Moretti has focused on historical periodization, as in Algee-Hewitt's examination of 3,439 plays looking only at the Gini Coefficient of each play's eigenvector centrality to track changes in ensemble casts from 1500 to 1920 (Algee-Hewitt 2017).

Our project focuses on a novel application, the classification of literary genre. When scaled up to a corpus covering a wider historical time span, our approach to genre could also provide insight on the historic periodization of literature.

Moretti also identifies that, in the application of SNA to literature, "methods for the automated extraction of network data (named entity recognition, co-reference resolution) and their evaluation are of particular importance," (Moretti 2011), which we accomplish in this thesis.

2.3 Gephi Toolkit

Gephi is an open source software for graph and network analysis, which allows for fast visualization and manipulation of large networks. As a generalist tool, "it provides easy and broad access to network data and allows for spatializing, filtering, navigating, manipulating and clustering" (Bastian, Heymann, and Jacomy 2009). Gephi also calculates a wide range of mathematical features for each graph, which we use as the basis for our mathematical analysis (as discussed in more detail in 3.3).

2.4 References

1) Algee-Hewitt, M. 2017. Distributed Character: Quantitative Models of the English Stage, 1500-1920. In Digital Humanities 2017: Book of Abstracts. Montreal: McGill University and Université de Montréal, pp. 119–21.

2) Bastian, M., S. Heymann, and M. Jacomy. 2009. Gephi: An Open Source Software for Exploring and Manipulating Networks. In International AAAI Conference on Web and Social Media, North America.

3) Armour, Cherie, Eiko, Marie K. Deserno, Jack Tsai, Robert H. Pietrzak. 2017. A network analysis of DSM-5 posttraumatic stress disorder symptoms and correlates in U.S. military veterans. In Journal of Anxiety Disorders, Volume 45, 2017, Pages 49-59.

4) Anggraini, D., S. Madenda, E. P. Wibowo and L. Boumedjout. 2015. Network Disintegration in Criminal Network. In 11th International Conference on Signal-Image Technology & Internet-Based Systems (SITIS), Bangkok, 2015, pp. 192-199.

5) Watts, D. 2001 "Small Worlds: The Dynamics of Networks between Order and Randomness", Princeton University Press.

6) Elson, D. K., N. Dames, and K. R. McKeown. 2010. Extracting Social Networks from Literary Fiction. In Proceedings of ACL 2010. Uppsala, pp. 138–47.

7) Moretti, F. 2011. Network Theory, Plot Analysis. New Left Review, 68:80–102.

8) Fischer F., M. Göbel, D. Kampkaspar, and P.Trilcke. 2015. Digital Network Analysis of Dramatic Texts. In Digital Humanities 2015 Conference Abstracts. University of Western Sydney.

9) Jannidis F., I. Reger, M. Krug, L. Weimer, L. Macharowsky, and F. Puppe. 2016. Comparison of Methods for the Identification of Main Characters in German Novels. In Digital Humanities Conference Abstracts, Jagiellonian University & Pedagogical University, Kraków, pp. 578–82.

10) Scott, J. 2000. "Social Network Analysis: A Handbook", 2nd ed., Sage Publications, London.

11) Shukla, Manisha, Susan Gauch and Lawrence Evalyn. 2018. Theatrical Genre Prediction Using Social Network Metrics. In 10th International Conference on Knowledge Discovery and Information Retrieval, Seville, Spain. (Accepted).

12) Park, G. M., S. H. Kim, and H. G. Cho. 2013. Structural Analysis on Social Network Constructed from Characters in Literature Texts. In Journal of Computers 8.9, pp. 2442–47.

13) Dreżewski, Rafał, Jan Sepielak, and Wojciech Filipkowski. 2015. The application of social network analysis algorithms in a system supporting money laundering detection. In Information Sciences, Volume 295, 2015, Pages 18-32, ISSN 0020-0255.

14) Trequattrini, Raffaele, Rosa Lombardi, Mirella Battista. 2015. Network analysis and football team performance: a first application. In Team Performance Management: An International Journal, Vol. 21 Issue: 1/2, pp.85-110.

15) Billah, S. M. and S. Gauch. 2015. Social network analysis for predicting emerging researchers. In 7th International Joint Conference on Knowledge Discovery, Knowledge Engineering and Knowledge Management (IC3K), Lisbon, 2015.

16) Wasserman, S. and K. Faust. 1994. “Social Network Analysis: Methods and Applications”, Cambridge University Press.

17) Serrat, O. 2017. Social network analysis. In Knowledge solutions (pp. 39-43). Springer, Singapore.

18) Stiller, J. and M. Hudson 2005. Weak Links and Scene Cliques Within the Small World of Shakespeare. In Journal of Cultural and Evolutionary Psychology 3, pp. 57–73.

19) Stiller, J., D. Nettle, and R. I. M. Dunbar.2003. The Small World of Shakespeare's Plays. In Human Nature, 14(4): 397–408.

20) Trilcke, P., F. Fischer, M. Göbel, and D. Kampkaspar. 2015. In 200 Years of Literary Network Data.

21) Trilcke, P., F. Fischer, M. Göbel, D. Kampkaspar, and C. Kittel. 2016. Theatre Plays as 'Small Worlds' Network Data on the History and Typology of German Drama, 1730–1930. In Digital Humanities 2016 Conference Abstracts. Jagiellonian University & Pedagogical University, Kraków, pp. 385–87.

22) Chuan-yi, Wang, Lv Xiao-hong, and Cao Yi. 2016. An empirical study on the collaboration of scholars in graduate education: based on the social network analysis. In Proceedings of the 2016 International Conference on Intelligent Information Processing (ICIIP '16). ACM, New York, NY, USA, Article 36, 7 pages.

3. DESIGN

The system for identifying genre consists of three building blocks: The Play Parser, the Social Network Generator and the Genre Predictor. Figure 5 shows the main components of the system architecture, which are discussed in more detail in the following subsections.

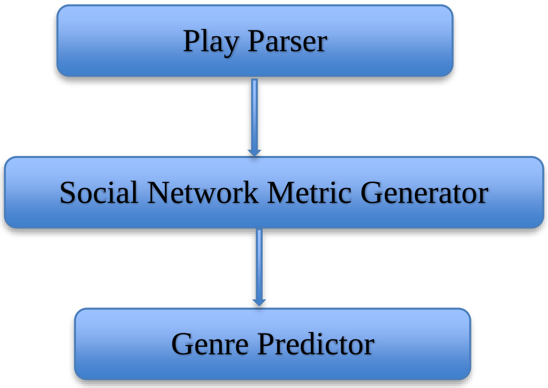


Figure 5: Block diagram of our system. (Shukla et.al. 2018)

3.1 Corpus

We will focus our work on the plays of William Shakespeare, one of the most widely studied authors of English literature. These plays have been digitized and manually encoded with Extensible Markup Language (XML) tags.

XML is a markup language that defines a set of rules for encoding documents in a format that is both human-readable and machine-readable. The design goals of XML emphasize simplicity, generality, and usability across the Internet. Although the design of XML focuses on documents, the language is widely used for the representation of arbitrary data structures such as those used in web services. [https://en.wikipedia.org/wiki/XML#cite_note-3]

The Text Encoding Initiative (TEI) is a consortium that collectively develops and maintains a standard for the representation of texts in digital form. Its chief deliverable is a set of Guidelines that specify encoding methods for machine-readable texts, mainly the ones in humanities, social sciences and linguistics. Since 1994, the TEI Guidelines have been widely used by libraries, museums, publishers, and individual scholars to present texts for online research, teaching, and preservation. [http://www.tei-c.org/index.xml].

For this project we downloaded the plays from the website http://showcases.exist-db.org/exist/apps/Showcases/index.html that has TEI-encoded XML formatted Shakespeare Plays. Although TEI provides tagging scheme, each organization has its own version of that scheme.

3.2 Play Parser

The main purpose of this component is to automatically parse TEI-encoded XML formatted plays to extract basic information such as the total number of characters, the name and role of each character, and the total number of acts and scenes in a play. For each scene, we used our parsed information to determine which characters were present in the scene (using stage directions to account for entrances and exits during a scene), and how many lines and words were spoken by each character.

```
<teiHeader>
      <fileDesc>
         <titleStmt>
            <title>All's Well That Ends Well</title>
            <author>William Shakespeare</author>
              ……..
</teiHeader>
```

Portion of the input file containing the Play title, which would be extracted from the corresponding node ‘title’.

Figure 6: Play Title extraction

We used java DOM API to parse XML files as it is designed to work very well with mixed content model and is not language dependent [https://blogs.oracle.com/thejavatutorials/jdom-and-dom4j-vs-dom]. In this library, all the plays were consistent in their tagging scheme. We designed a parser that takes XML-formatted play as input, parses the file and stores the relevant tag information into Character objects. Figure 6,7,8 and 9 shows what information is contained in the file followed by how it is extracted.

```
<text>
    <front>
        <div xml:id="sha-awwcast" type="castList">
            <head>Dramatis Personae</head>
            <castList>
                <castItem type="role">
                    <role xml:id="FranceK">King of France</role>
                </castItem>
                <castItem type="role">
                    <role xml:id="FlorenceD">Duke of Florence</role>
                </castItem>
                <castItem type="role">
                    <role xml:id="Bertram">Bertram</role>
                    <roleDesc>Count of Rousillon</roleDesc>
                </castItem>
                ……….
            </castList>
```

In this portion of the file, we iterate through the node ‘castList’ and fetch the cast list of the play. This contains information about role name and role description of each Character in the play. It is worth noting that some of the characters mentioned in the cast list never appear in any play scenes or acts. We removed such characters from the list.

Figure 7: Character List information extraction

```
<div xml:id="sha-aww1">
            <head>Act 1</head>
            <div xml:id="sha-aww101">
               <head>Act 1, Scene 1</head>
               ……………
</div>
```

From this we fetched Act and Scene information. Each Act is associated with a specific id which is actually initials of the play appended with act number which in the example above is Act 1so, aww1.This is preceded by “sha-” Each scene is associated with an id as well which is a combination of act id followed by scene number, hence “sha-aww101”. Fetching act and scene by id is important so that we always process the correct information during parsing.

Figure 8: Act and Scene information extraction

```
<sp who="FranceK">
                 <speaker>King</speaker>
                 <l xml:id="sha-aww102001" n="1">
                       The Florentines and Senoys are by the ears;
                  </l>
                 <l xml:id="sha-aww102002" n="2">
                       Have fought with equal fortune and continue
                 </l>
</sp>
<sp who="aww-stew.">
                 <speaker>Steward</speaker>
                 <ab xml:id="sha-aww103100" n="100">
                      Madam, I was very late more near her than I think
                 </ab>
</sp>
```

In this example, the <sp> tag contains information about current speaker whereas the <who> tag encodes either who is speaking or who are present on stage. The <speaker> tag has the information about who is currently speaking. The <speaker> tag is always proceeded by either <l> tag or <ab> tag to represent what lines were spoken by the person currently speaking. We parsed information inside <l> and <ab> tag to count the number of line spoken. While counting the line, we also counted the words by splitting the line by space.

Figure 9: Speaker, Line and Words extraction

The parsed information is stored in a Character object that has attributes for the total number of lines and words spoken by that character in the play along with the total number of acts and scenes in which it appeared, as mentioned in Figure 10. After building the Character object, we pass the list of characters to next component, Social Network Metric Calculator. The information extracted, forms the play feature component of features used in genre prediction as mentioned in Table 1.

```
public class Character {

   String Name;
   ArrayList<String> acts_scenes;
   int no_of_lines_spoken;
   int no_of_words_spoken;
   HashMap<String, ArrayList<Integer>> sceneInfo;
….
}
```

1. Name - stores name of the Character.
2. Acts_Scenes – stores information about act number and scene number in which the character appeared.
3. no_of_lines_spoken – stores information about total number of lines spoken by the character in the play.
4. no_of_words_spoken - stores information about total number of words spoken by the character in the play.
5. sceneInfo - maps scenes to number of words spoken by the character in that scene.

Figure 10: Java Character Object

3.2 Social Network Metric Calculator

This component creates each play’s social network graph using the information generated by the Play Parser described in Section 3.1 and then calculates social network features from the generated graph of the play. We used Gephi’s API to generate the graph files. The character list from the Play Parser is passed to the Social Network Metric Calculator. Each character in the list maps to a graph node using Gephi API graph module. Once we have all the nodes identified, we use the scene information for each character to create an edge between this character and the remaining characters in the list, if they both appeared in the same scene.

We created two types of graphs. For non-directional graphs, we summed the total number of words spoken by each character in the shared scenes to determine the edge weights. For directional graphs, we created a directed edge from Character 1 to Character 2, weighted by the number of words spoken by Character 1 in the shared scenes. We also added another directed edge from Character 2 to Character 1, weighted by the number of words spoken by Character 2 in the shared scenes. The resulting graphs for each play are available online at http://text.csce.uark.edu:8080/SocialNetworkOfShakespearePlays/.

Once the basic structure of graph is ready, we computed node and graph metrics using functions provided by Gephi. In total, 17 graph metrics were calculated; these are presented as network features in Table 1.

Table 1: Features extracted from Shakespeare's plays. Here *g* represents a graph for a specific play, *c* a character node in the graph, and *e* an edge between two character nodes in the graph.

Extracted Features
1. tot_characters = total number of characters of g
2. tot_edges = total number of edges of g
3. tot_lines = total number of lines spoken by c in n
4. tot_words = total number of words spoken by c in n

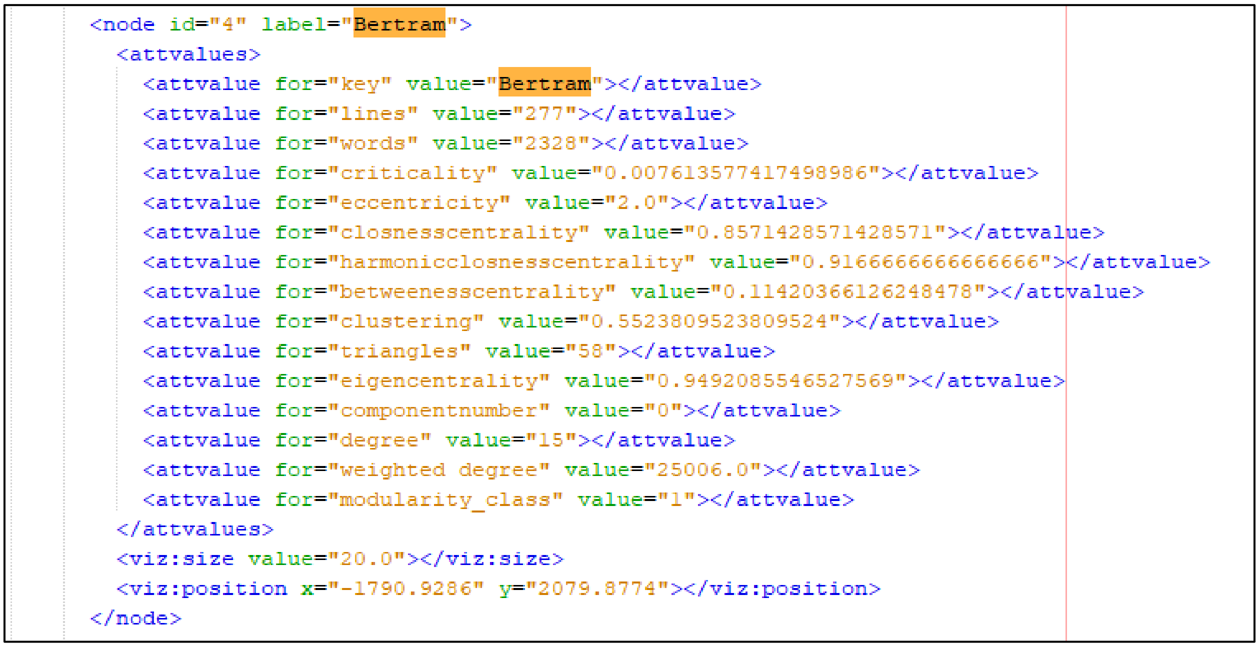
Table 1: Features extracted from Shakespeare's plays. Here *g* represents a graph for a specific play, *c* a character node in the graph, and *e* an edge between two character nodes in the graph (Cont.)

Network Features
Node Features
5. Degree = set of adjacent nodes of c in the graph
6. Criticality = A k-critical graph is a critical graph with chromatic number k; a graph G with chromatic number k is k-vertex-critical if each of its vertices is a critical element.
7. Eigenvector Centrality = A measure of c's importance in a network based on c's connections.
8. Eccentricity = The eccentricity of a graph vertex in a connected graph is the maximum graph distance between and any other vertex of.
9. Closeness Centrality = The average distance from a given node to all other nodes in the network.
10. Harmonic Centrality = In a (not necessarily connected) graph, the harmonic centrality reverses the sum and reciprocal operations in the definition of closeness centrality.
11. Betweenness Centrality = Node Betweenness Centrality measures how often a node appears on shortest paths between nodes in the network.
12. Weighted Degree = weighted degree of a node is based on the number of edge for a node, but ponderated by the weight of each edge. It's doing the sum of the weight of the edges.
Graph Features
13. Clustering Coefficient = The clustering coefficient, when applied to a single node, is a measure of how complete the neighborhood of a node is. When applied to an entire network, it is the average clustering coefficient over all nodes in the network.
14. Density = Measures how close the network is to complete. A complete graph has all possible edges and density equal to 1. For undirected graph, density is equal to $(2*\lvert e\rvert) / (\lvert c\rvert(\lvert c\rvert-1))$. For directed graph, it is $\lvert e\rvert / (\lvert c\rvert(\lvert c\rvert-1))$.
15. Diameter = The maximal distance between all pairs of nodes.
16. Path Length = The average graph-distance between all pairs of nodes.
17. Connected Components = A connected component of an undirected graph is a maximal set of nodes such that each pair of nodes is connected by a path.
18. Modularity = Measures how well a network decomposes into modular communities.
19. Average Degree = Sum of the degrees of all the nodes in the graph divided by the total number of nodes in the graph.
20. Average Weighted Degree = Sum of the degrees of all the nodes in the graph divided by the total number of nodes in the graph.
21. Radius = The radius of a graph is the minimum graph eccentricity of any graph vertex in a graph.

The input to the calculator is Character List and the output is gexf file and csv file. We chose gexf format as with this file format literary scholars can directly import this file into GEPHI tool to perform various analysis on the graph. This file is also used to display the graph on the website.

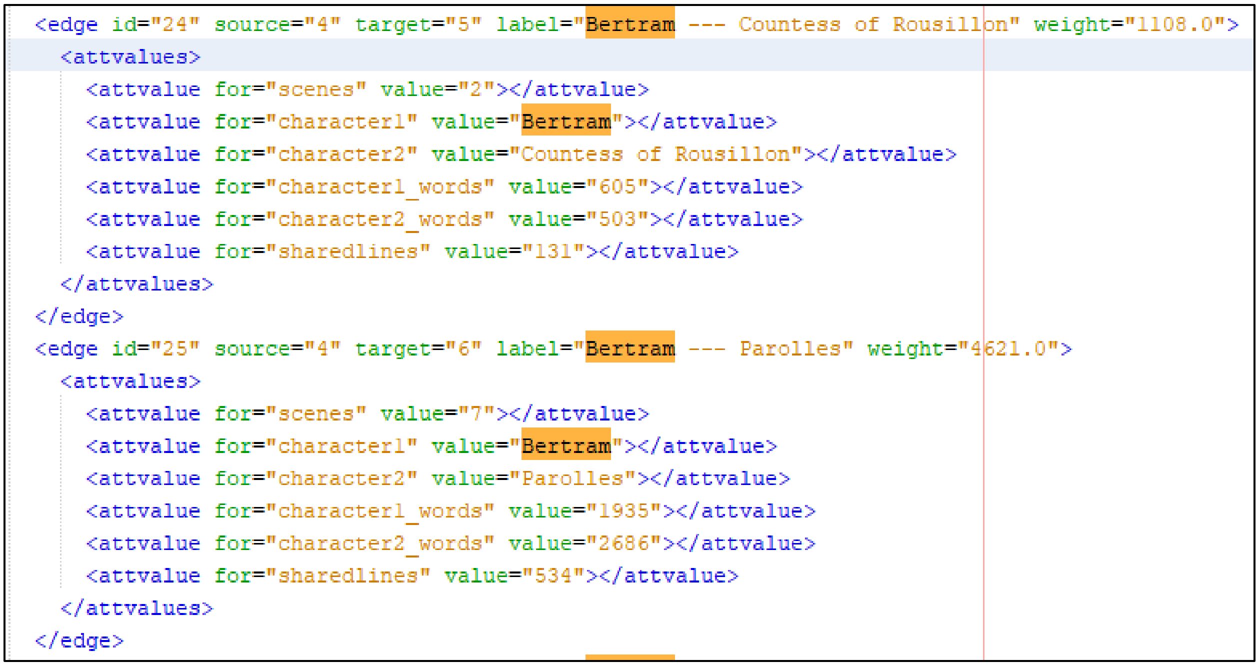
Input
List of Characters { c1, c2, c3,…. }
Where c1, c2 and c3 are Character objects as mentioned in Figure 10.

Figure 11: Input to Social Network Metric Calculator



```
<node id="4" label="Bertram">
  <attvalues>
    <attvalue for="key" value="Bertram"></attvalue>
    <attvalue for="lines" value="277"></attvalue>
    <attvalue for="words" value="2328"></attvalue>
    <attvalue for="criticality" value="0.007613577417498986"></attvalue>
    <attvalue for="eccentricity" value="2.0"></attvalue>
    <attvalue for="closnesscentrality" value="0.8571428571428571"></attvalue>
    <attvalue for="harmonicclosnesscentrality" value="0.9166666666666666"></attvalue>
    <attvalue for="betweenesscentrality" value="0.11420366126248478"></attvalue>
    <attvalue for="clustering" value="0.5523809523809524"></attvalue>
    <attvalue for="triangles" value="58"></attvalue>
    <attvalue for="eigencentrality" value="0.9492085546527569"></attvalue>
    <attvalue for="componentnumber" value="0"></attvalue>
    <attvalue for="degree" value="15"></attvalue>
    <attvalue for="weighted degree" value="25006.0"></attvalue>
    <attvalue for="modularity_class" value="1"></attvalue>
  </attvalues>
  <viz:size value="20.0"></viz:size>
  <viz:position x="-1790.9286" y="2079.8774"></viz:position>
</node>
```

Figure 12: Part of gexf file (Node representation)– Sample Output from Social Network Metric Calculator

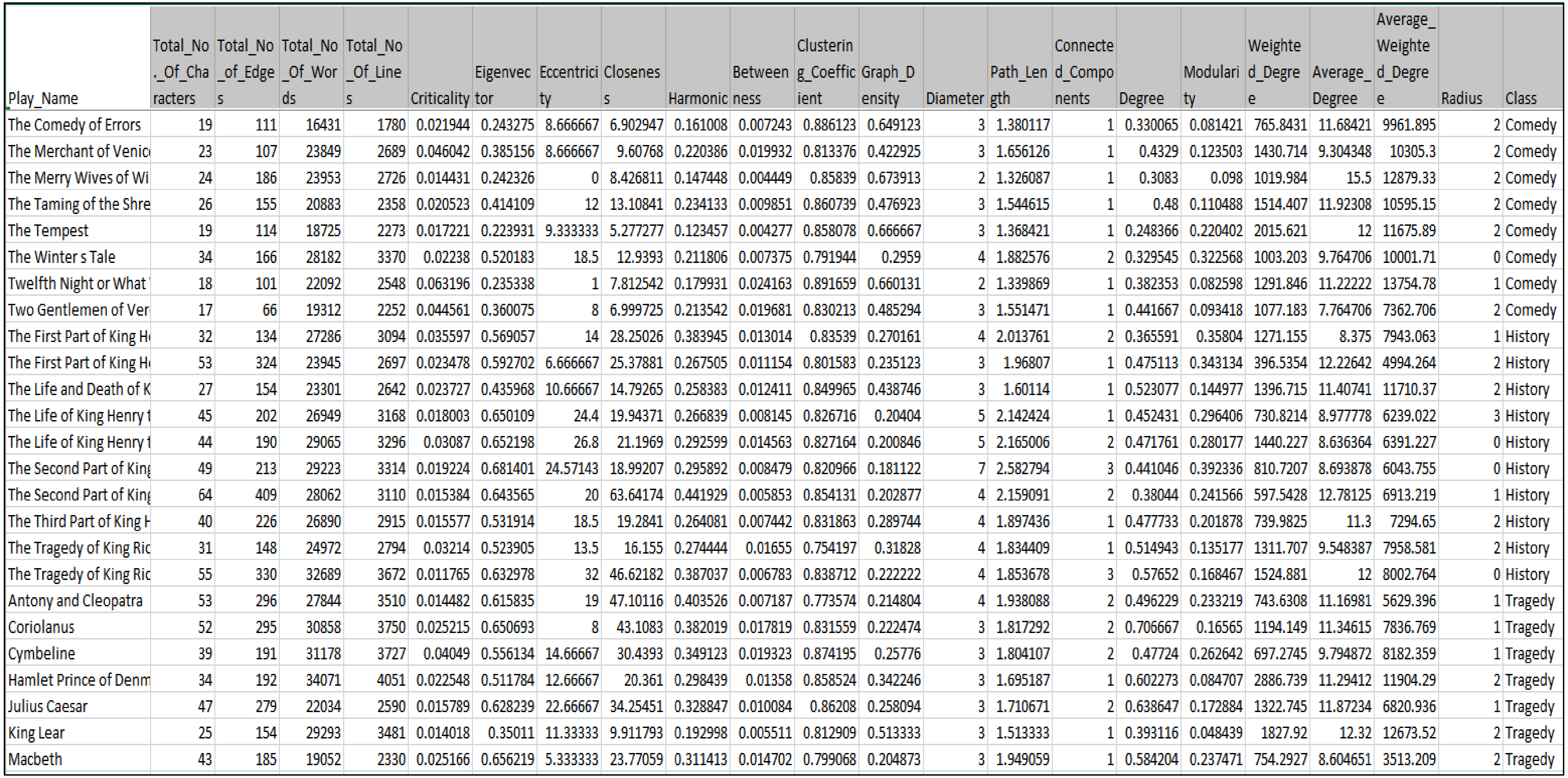


```
<edge id="24" source="4" target="5" label="Bertram --- Countess of Rousillon" weight="1108.0">
  <attvalues>
    <attvalue for="scenes" value="2"></attvalue>
    <attvalue for="character1" value="Bertram"></attvalue>
    <attvalue for="character2" value="Countess of Rousillon"></attvalue>
    <attvalue for="character1_words" value="605"></attvalue>
    <attvalue for="character2_words" value="503"></attvalue>
    <attvalue for="sharedlines" value="131"></attvalue>
  </attvalues>
</edge>
<edge id="25" source="4" target="6" label="Bertram --- Parolles" weight="4621.0">
  <attvalues>
    <attvalue for="scenes" value="7"></attvalue>
    <attvalue for="character1" value="Bertram"></attvalue>
    <attvalue for="character2" value="Parolles"></attvalue>
    <attvalue for="character1_words" value="1935"></attvalue>
    <attvalue for="character2_words" value="2686"></attvalue>
    <attvalue for="sharedlines" value="534"></attvalue>
  </attvalues>
</edge>
```

Figure 12: Part of gexf file (Edge representation)– Sample Output from Social Network Metric Calculator cont.

CSV files contain calculated metrics from each play in a single file which is used as an input for Genre Predictor module. See figure 11, 12, 13 for input to and output from Social Network Metric Calculator. Generating all the gexf and csv files took approximately 3 seconds on MAC OS system with 8 GB RAM.

We have made the network graphs and selected mathematical features available online at http://text.csce.uark.edu:8080/SocialNetworkOfShakespearePlays/. Figure 14 shows a screen shot of Hamlet, the first play we analyzed. Figure 15 shows Hamlet after the user has interacted with the play to rearrange the character node placements. Users can click on a character to see more information about the node features (see Figure 16). Users can click on an edge to see more information about the edge features (see Figure 17).



Play_Name	Total_No._Of_Characters	Total_No_of_Edges	Total_No_Of_Words	Total_No_Of_Lines	Criticality	Eigenvector	Eccentricity	Closeness	Harmonic	Betweenness	Clustering_Coefficient	Graph_Density	Diameter	Path_Length	Connected_Components	Degree	Modularity	Weighted_Degree	Average_Degree	Average_Weighted_Degree	Radius	Class
The Comedy of Errors	19	111	16431	1780	0.021944	0.243275	8.666667	6.902947	0.161008	0.007243	0.886123	0.649123	3	1.380117	1	0.330065	0.081421	765.8431	11.68421	9961.895	2	Comedy
The Merchant of Venic	23	107	23849	2689	0.046042	0.385156	8.666667	9.60768	0.220386	0.019932	0.813376	0.422925	3	1.656126	1	0.4329	0.123503	1430.714	9.304348	10305.3	2	Comedy
The Merry Wives of Wi	24	186	23953	2726	0.014431	0.242326	0	8.426811	0.147448	0.004449	0.85839	0.673913	2	1.326087	1	0.3083	0.098	1019.984	15.5	12879.33	2	Comedy
The Taming of the Shre	26	155	20883	2358	0.020523	0.414109	12	13.10841	0.234133	0.009851	0.860739	0.476923	3	1.544615	1	0.48	0.110488	1514.407	11.92308	10595.15	2	Comedy
The Tempest	19	114	18725	2273	0.017221	0.223931	9.333333	5.277277	0.123457	0.004277	0.858078	0.666667	3	1.368421	1	0.248366	0.220402	2015.621	12	11675.89	2	Comedy
The Winter s Tale	34	166	28182	3370	0.02238	0.520183	18.5	12.9393	0.211806	0.007375	0.791944	0.2959	4	1.882576	2	0.329545	0.322568	1003.203	9.764706	10001.71	0	Comedy
Twelfth Night or What	18	101	22092	2548	0.063196	0.235338	1	7.812542	0.179931	0.024163	0.891659	0.660131	2	1.339869	1	0.382353	0.082598	1291.846	11.22222	13754.78	1	Comedy
Two Gentlemen of Ver	17	66	19312	2252	0.044561	0.360075	8	6.999725	0.213542	0.019681	0.830213	0.485294	3	1.551471	1	0.441667	0.093418	1077.183	7.764706	7362.706	2	Comedy
The First Part of King H	32	134	27286	3094	0.035597	0.569057	14	28.25026	0.383945	0.013014	0.83539	0.270161	4	2.013761	2	0.365591	0.35804	1271.155	8.375	7943.063	1	History
The First Part of King H	53	324	23945	2697	0.023478	0.592702	6.666667	25.37881	0.267505	0.011154	0.801583	0.235123	3	1.96807	1	0.475113	0.343134	396.5354	12.22642	4994.264	2	History
The Life and Death of K	27	154	23301	2642	0.023727	0.435968	10.66667	14.79265	0.258383	0.012411	0.849965	0.438746	3	1.60114	1	0.523077	0.144977	1396.715	11.40741	11710.37	2	History
The Life of King Henry t	45	202	26949	3168	0.018003	0.650109	24.4	19.94371	0.266839	0.008145	0.826716	0.20404	5	2.142424	1	0.452431	0.296406	730.8214	8.977778	6239.022	3	History
The Life of King Henry t	44	190	29065	3296	0.03087	0.652198	26.8	21.1969	0.292599	0.014563	0.827164	0.200846	5	2.165006	2	0.471761	0.280177	1440.227	8.636364	6391.227	0	History
The Second Part of King	49	213	29223	3314	0.019224	0.681401	24.57143	18.99207	0.295892	0.008479	0.820966	0.181122	7	2.582794	3	0.441046	0.392336	810.7207	8.693878	6043.755	0	History
The Second Part of King	64	409	28062	3110	0.015384	0.643565	20	63.64174	0.441929	0.005853	0.854131	0.202877	4	2.159091	2	0.38044	0.241566	597.5428	12.78125	6913.219	1	History
The Third Part of King H	40	226	26890	2915	0.015577	0.531914	18.5	19.2841	0.264081	0.007442	0.831863	0.289744	4	1.897436	1	0.477733	0.201878	739.9825	11.3	7294.65	2	History
The Tragedy of King Ric	31	148	24972	2794	0.03214	0.523905	13.5	16.155	0.274444	0.01655	0.754197	0.31828	4	1.834409	1	0.514943	0.135177	1311.707	9.548387	7958.581	2	History
The Tragedy of King Ric	55	330	32689	3672	0.011765	0.632978	32	46.62182	0.387037	0.006783	0.838712	0.222222	4	1.853678	3	0.57652	0.168467	1524.881	12	8002.764	0	History
Antony and Cleopatra	53	296	27844	3510	0.014482	0.615835	19	47.10116	0.403526	0.007187	0.773574	0.214804	4	1.938088	2	0.496229	0.233219	743.6308	11.16981	5629.396	1	Tragedy
Coriolanus	52	295	30858	3750	0.025215	0.650693	8	43.1083	0.382019	0.017819	0.831559	0.222474	3	1.817292	2	0.706667	0.16565	1194.149	11.34615	7836.769	1	Tragedy
Cymbeline	39	191	31178	3727	0.04049	0.556134	14.66667	30.4393	0.349123	0.019323	0.874195	0.25776	3	1.804107	2	0.47724	0.262642	697.2745	9.794872	8182.359	1	Tragedy
Hamlet Prince of Denm	34	192	34071	4051	0.022548	0.511784	12.66667	20.361	0.298439	0.01358	0.858524	0.342246	3	1.695187	1	0.602273	0.084707	2886.739	11.29412	11904.29	2	Tragedy
Julius Caesar	47	279	22034	2590	0.015789	0.628239	22.66667	34.25451	0.328847	0.010084	0.86208	0.258094	3	1.710671	2	0.638647	0.172884	1322.745	11.87234	6820.936	1	Tragedy
King Lear	25	154	29293	3481	0.014018	0.35011	11.33333	9.911793	0.192998	0.005511	0.812909	0.513333	3	1.513333	1	0.393116	0.048439	1827.92	12.32	12673.52	2	Tragedy
Macbeth	43	185	19052	2330	0.025166	0.656219	5.333333	23.77059	0.311413	0.014702	0.799068	0.204873	3	1.949059	1	0.584204	0.237471	754.2927	8.604651	3513.209	2	Tragedy

Figure 13: Part of CSV file – Sample CSV Output from Social Network Metric Calculator

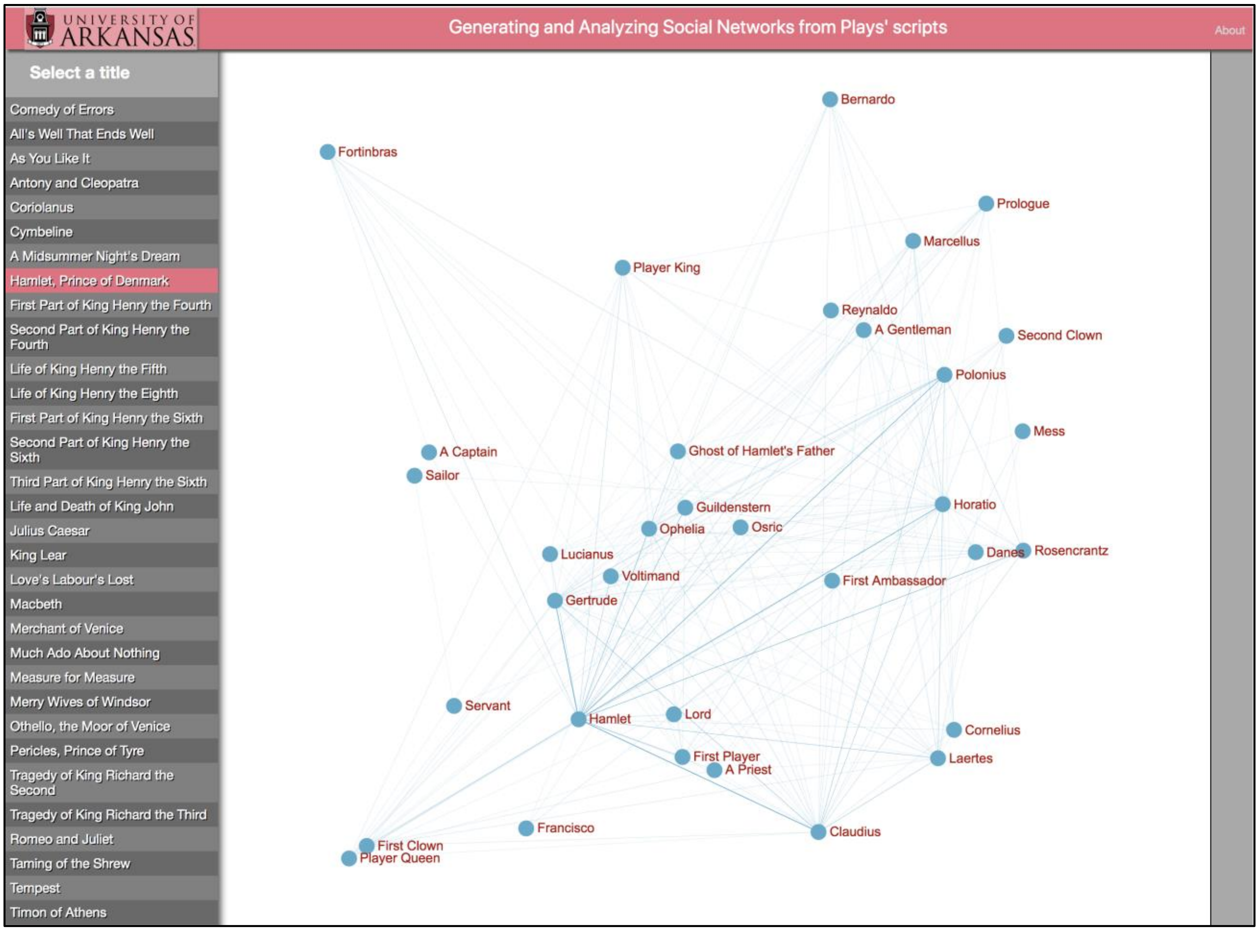


Figure 14: Hamlet Social Network

3.2.1 Extracted Features

Some features we studied were extracted from the play itself, i.e., not generated by the social network, e.g., total number of characters in the play (see Table 2). As our results in 4.3.1 and 4.3.3 demonstrate, despite their simplicity as features, the number of edges and the number of words spoken in a play can play a crucial role in identifying the genre.

3.2.2 Network Features

We compute the network features of the graph using Gephi’s library. Node Features such as Eigenvector capture information about a particular node in the graph/character in the play. In contrast Graph Features such as Path Length capture information about the graph/play as a whole.

For the Node Features, we normalized the values using by calculating the network centralized value using the following network level centralization index as mentioned by Newman (Newman 2010):

$$\mathrm{C} = \frac{\sum_{\mathrm{i}}[\mathrm{c}^{*} - \mathrm{c}^{\mathrm{i}}]}{\mathrm{Max}\sum_{\mathrm{i}}[\mathrm{c}^{*} - \mathrm{c}^{\mathrm{i}}]}$$

where,

c* = maximum value for all the nodes in the graph

ci = value of current node

And in denominator, maximum of the summation over all the possible networks. This method helps in converting node metrics into graph metrics for evaluation purpose.

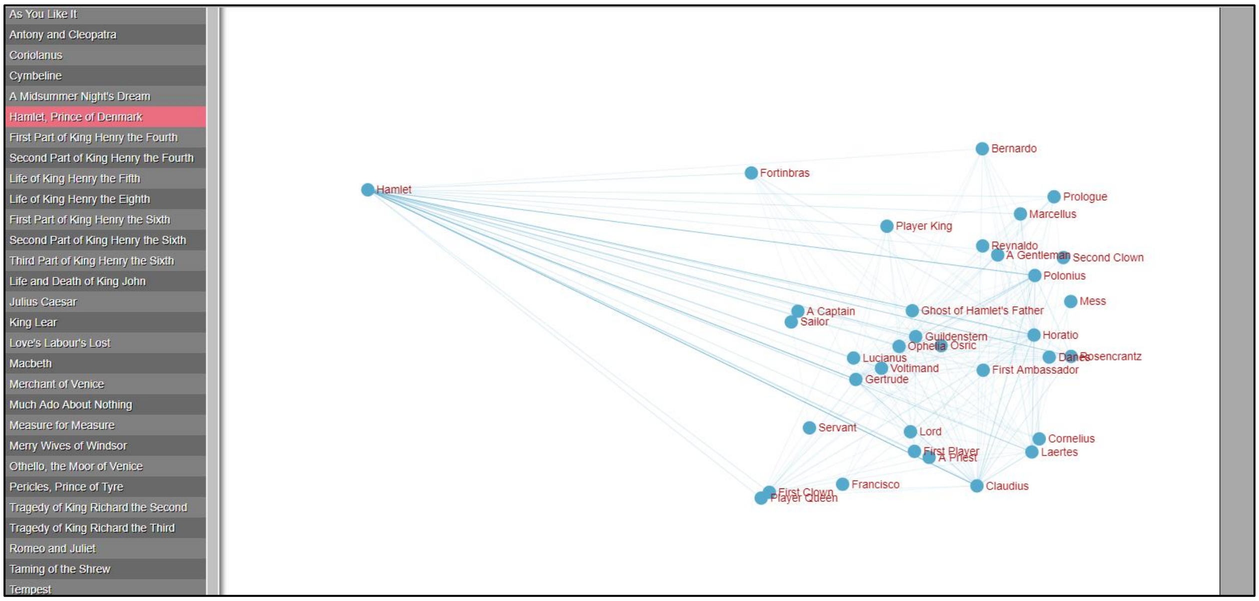
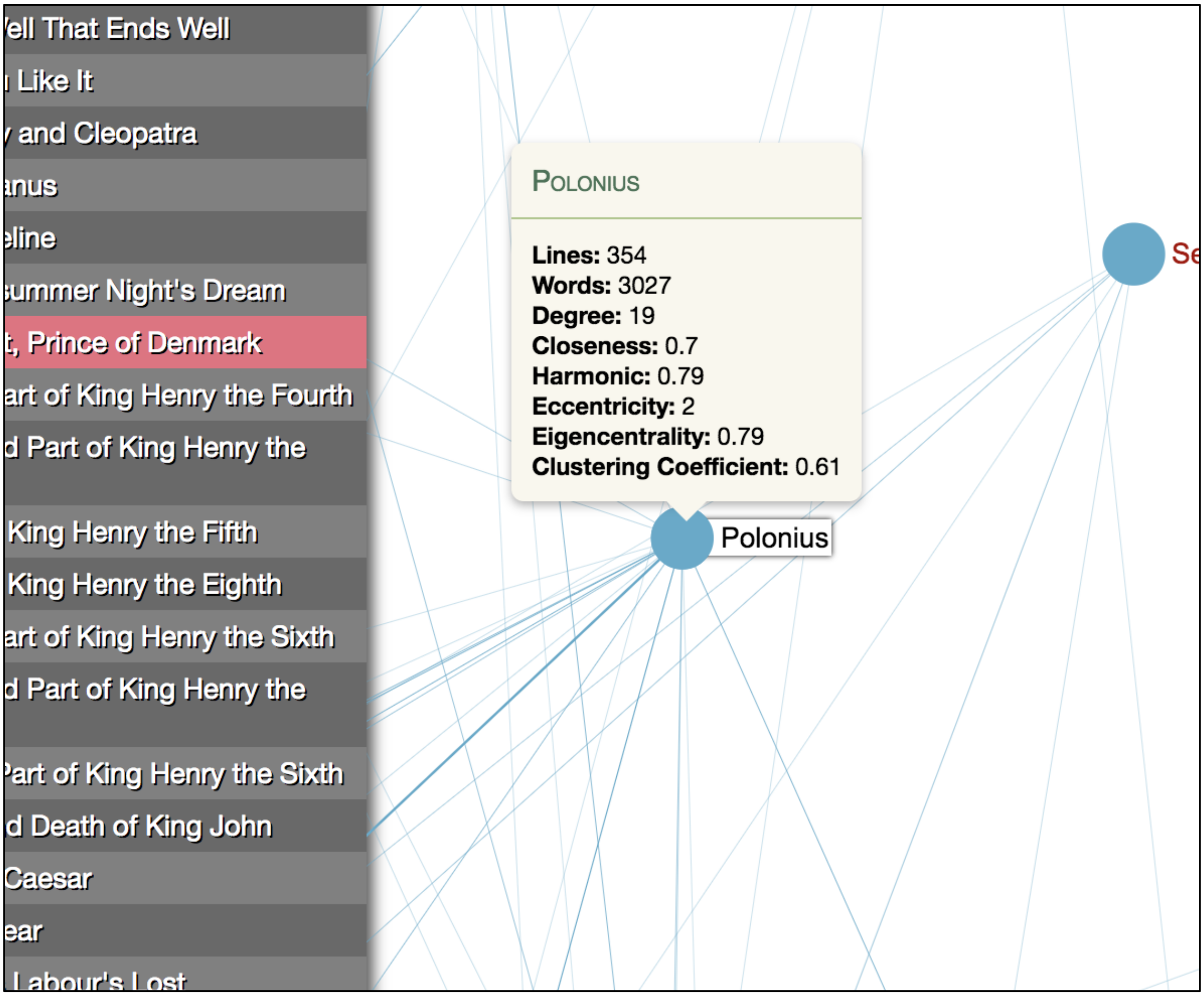


Figure 15: User Interaction with Hamlet Network



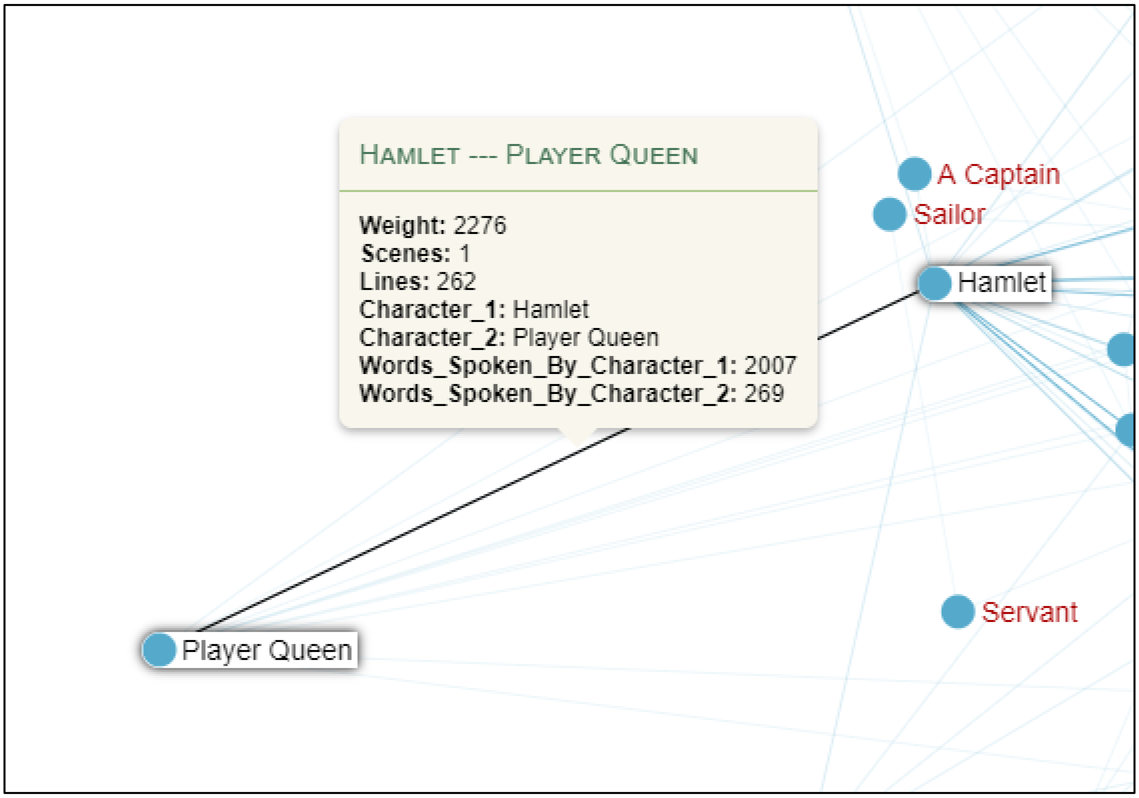
POLONIUS

Lines: 354
Words: 3027
Degree: 19
Closeness: 0.7
Harmonic: 0.79
Eccentricity: 2
Eigencentrality: 0.79
Clustering Coefficient: 0.61

Figure 16: Character Information

3.3 Genre Predictor

This module is the key to our research. Given a set of training plays by Shakespeare labeled as Tragedy, Comedy, or History, it predicts the genre of a testing (previously unseen) Shakespeare play. This module trains the predictor using a subset of the features extracted above on a set of labeled plays. This is essentially a classification process that has been widely studied in Machine Learning. Three of the most widely used classifiers are K-Nearest Neighbor (Aha and Kibler 1991), Support Vector Machines (Chang and Lin 2011) and Naïve Bayes (John and Langley 1995).



Hamlet --- Player Queen

Weight: 2276
Scenes: 1
Lines: 262
Character_1: Hamlet
Character_2: Player Queen
Words_Spoken_By_Character_1: 2007
Words_Spoken_By_Character_2: 269

Figure 17: Edge Information

As reported in Manning (Manning, Raghavan and Schutze 2008), if the training set is small (Forman and Cohen 2004; Klein and Manning 2002), high bias/low variance classifiers (e.g., Naive Bayes) have an advantage over low bias/high variance classifiers (e.g., KNN), since the latter will overfit. But low bias/high variance classifiers start to win out as the training set grows (they have lower asymptotic error), since high bias classifiers are not powerful enough to provide accurate models. As mentioned by Forman (Forman and Cohen 2004), in case of little data to train a supervised classifier, machine learning theory recommends selecting a classifier with high bias. For example, there are theoretical and empirical results showing that Naive Bayes does well in such circumstances (Forman and Cohen 2004; Ng and Jordan 2001), although this effect is not necessarily observed in practice with regularized models over textual data (Klein and Manning 2002). At any rate, a very low bias model like a nearest neighbor model is probably contraindicate. (https://nlp.stanford.edu/IR-book/html/htmledition/choosing-what-kind-of-classifier-to-use-1.html).

Support Vector Machines (SVMs) also work well with limited data. High accuracy, nice theoretical guarantees regarding overfitting. SVMs) are a popular machine learning method for classification, regression, and other learning tasks. Since our classification problem had more than two classes, we combined SVM with One vs One (OvO) classification. This works as follows: choose a pair of classes from a set of *n* classes, which in our case is three (comedy, history and tragedy) and develop a binary classifier for each pair. Create all possible combinations of pairs of classes from *n* and then for each pair develop a binary SVM. The final class is assigned to each unseen play based on the class chosen by maximum number of binary SVM classifiers. By using OvO, our SVM is much less sensitive to the problems of imbalanced datasets, which is particularly helpful given the different sizes of each of our three classes and our small overall sample size (Chang and Lin 2011). In chapter 4, we evaluate each of the three classifiers above to see which works best for our application.

3.4 References

1) Chang, Chih-Chung and Chih-Jen Lin. 2011. LIBSVM: A library for support vector machines. ACM Trans. In Intell. Syst. Technol. 2, 3, Article 27 (May 2011), 27 pages.

2) Manning, Christopher D., Prabhakar Raghavan and Hinrich Schütze. 2008. “Introduction to Information Retrieval”. Cambridge University Press.

3) Aha, D. and D. Kibler. 1991. Instance-based learning algorithms. Machine Learning. 6:37-66.

4) Forman, George, and Ira Cohen. 2004. Learning from little: Comparison of classifiers given little training. In Proc. PKDD, pp. 161-172.

5) John, George H., and Pat Langley. 1995. Estimating Continuous Distributions in Bayesian Classifiers. In: Eleventh Conference on Uncertainty in Artificial Intelligence, San Mateo, 338-345.

6) http://showcases.exist-db.org/exist/apps/Showcases/index.html

7) http://text.csce.uark.edu:8080/SocialNetworkOfShakespearePlays/

8) http://www.tei-c.org/index.xml

9) https://blogs.oracle.com/thejavatutorials/jdom-and-dom4j-vs-dom

10) https://en.wikipedia.org/wiki/XML#cite_note-3

11) https://nlp.stanford.edu/IR-book/html/htmledition/choosing-what-kind-of-classifier-to-use-1.html

12) Klein, Dan, and Christopher D. Manning. 2002. Conditional structure versus conditional estimation in NLP models. In Proc. Empirical Methods in Natural Language Processing, pp. 9-16.

13) Shukla, Manisha, Susan Gauch and Lawrence Evalyn. 2018. Theatrical Genre Prediction Using Social Network Metrics. In 10th International Conference on Knowledge Discovery and Information Retrieval, Seville, Spain. (Accepted).

14) Newman, M.E.J. 2010. Networks: An Introduction. Oxford, UK: Oxford University Press.

15) Ng, Andrew Y., and Michael I. Jordan. 2001. On discriminative vs. generative classifiers: A comparison of logistic regression and naive Bayes. In Proc. NIPS, pp. 841-848. URL: www-2.cs.cmu.edu/Groups/NIPS/NIPS2001/papers/psgz/AA28.ps.gz.

4. EXPERIMENTS

4.1 Dataset

Our dataset is comprised of 36 plays by Shakespeare, in TEI encoded XML files. The dataset was downloaded from the website exist-db.org. We split dataset into five subsets, evenly balancing each genre in each subset. These were then used to perform five-fold cross validation to generate the results. Table 2 shows the list of plays used with their associated genres. There is some debate amongst scholars as to the genre of a few of the plays. There is some debate among literary scholars as to the genre of a few plays, but we used the most commonly agreed upon classification for each play.

4.2 Experimental Setup

Our generated network graphs were then used to test our central question: whether the social network of characters in a play can be used as a proxy for features of the play's narrative content. Can we use social network metrics to distinguish between the dramatic genres of tragedy, comedy, and history? We used the 21 different features listed in Table 1 to test our hypothesis. We first evaluated three classifiers to wee which worked best on our dataset. We then evaluated whether unidirectional links between nodes or bidirectional links (that capture who talks to whom) were better for genre prediction. Then, we investigated how well individual features were able to predict genre followed by evaluating the effectiveness of predictors trained on all combinations of pairs of features. We went on to evaluate combinations of larger sets to see if adding on more features increase accuracy of classifier's genre prediction. Finally, we explored the results of using our best classifier for literary purposes, e.g., predicting the genre of plays whose classifications are disputed. Section 4.3 discusses the result of each test.

Table 2: Dataset.

Play_Name	**Class**
All’s Well That Ends Well	Comedy
As You Like It	Comedy
A Midsummer Night’s Dream	Comedy
Love’s Labour’s Lost	Comedy
Measure for Measure	Comedy
Much Ado About Nothing	Comedy
The Comedy of Errors	Comedy
The Merchant of Venice	Comedy
The Merry Wives of Windsor	Comedy
The Taming of the Shrew	Comedy
The Tempest	Comedy
The Winter’s Tale	Comedy
Twelfth Night or What You Will	Comedy
Two Gentlemen of Verona	Comedy
The First Part of King Henry the Fourth	History
The First Part of King Henry the Sixth	History
The Life and Death of King John	History
The Life of King Henry the Eighth	History
The Life of King Henry the Fifth	History
The Second Part of King Henry the Fourth	History
The Second Part of King Henry the Sixth	History
The Third Part of King Henry the Sixth	History
The Tragedy of King Richard the Second	History
The Tragedy of King Richard the Third	History
Antony and Cleopatra	Tragedy
Coriolanus	Tragedy
Cymbeline	Tragedy
Hamlet Prince of Denmark	Tragedy
Julius Caesar	Tragedy
King Lear	Tragedy
Macbeth	Tragedy
Othello the Moor of Venice	Tragedy
Romeo and Juliet	Tragedy
Timon of Athens	Tragedy
Titus Andronicus	Tragedy
Troilus and Cressida	Tragedy

4.2.1 Classifier Selection

We performed five-fold cross validation training the classifiers using all the features mentioned in Table 1 and calculated accuracy with Weka API implementations of KNN, SVM and Naïve Bayes classifiers [https://weka.wikispaces.com] for genre prediction. All the experiments were conducted on MAC OS with 8 GB RAM. Table 3 shows that Naïve Bayes performed best out of the three when all the features were taken into consideration for classification. It is worth noting that because there are three genres, a random predictor would only have 33.33% accuracy. So, although 66.43% might seem like low accuracy for genre prediction, it is roughly two times more accurate than random guess.

Table 3: 5-fold cross-validation result of classification using all the features

Classifier	Accuracy
Naïve Bayes	66.43%
SVM with OVO	57.50%
KNN	48.93%

The biggest difference between the models from a features point of view is that Naive Bayes treats each feature as independent, whereas SVM looks at the interactions between the features to a certain degree when using a non-linear kernel. Since our features are likely to be non-independent, e.g., the number of words in a play are likely correlated with the number of lines in a play, we decided to present results for the following experiments using Naïve Bayes and SVM. Naïve Bayes was chosen because it was the best performing classifier in this initial testing; SVM was chosen because it might work better with smaller subsets of correlated features. KNN was eliminated from further consideration.

4.2.2 Graph Selection

After choosing the above two classifiers to conduct experiments, our next question was whether a directional or non-directional graph data is a better representation of the play information. Table 5 shows the calculated average value for each network metric per genre. It is observed that the graph metrics have the same average for directional or non-directional graphs. However, metrics involving edges vary for the two as shown in Table 4.

Table 4: Average of a metric for each genre – directional and non-directional

	Non-Directional			**Directional**		
Attribute	**Comedy**	**History**	**Tragedy**	**Comedy**	**History**	**Tragedy**
Total No of Characters	23.14	44	38.33	23.14	44	38.33
Total No of Edges	132	233	217.75	132	233	217.75
Total No of Words	22426.42	27238.2	27050.58	22426.42	27238.2	27050.58
Total No of Lines	2586.5	3070.2	3215	2586.5	3070.2	3215
Criticality	0.03	0.022	0.020	0.011	0.005	0.006
Eigenvector	0.34	0.59	0.52	0.79	0.84	0.81
Eccentricity	8.63	19.11	13.01	18.35	42.34	35.375
Closeness	9.28	27.42	24.95	13.09	34.33	27.94
Harmonic	0.19	0.31	0.29	0.25	0.34	0.32
Betweenness	0.012	0.010	0.011	0.004	0.002	0.0023
Clustering Coefficient	0.84	0.82	0.84	0.82	0.79	0.81
Graph Density	0.52	0.25	0.34	0.26	0.13	0.17
Diameter	2.85	4.3	3.08	2.93	3.9	3.42
Path Length	1.52	2.02	1.71	1.38	1.63	1.55
Connected Components	1.07	1.7	1.5	1.07	1.7	1.5
Degree	0.38	0.47	0.52	0.38	0.47	0.52
Modularity	0.14	0.25	0.16	0.15	0.26	0.16
Weighted Degree	1306.86	1022.029	1457.85	1306.86	1022.029	1457.85
Average Degree	11.31	10.39	11.38	11.31	10.39	11.38
Average Weighted Degree	11353.31	7349.09	9136.53	11353.31	7349.09	9136.53
In degree	NA	NA	NA	428.54	232.28	318.96
Outdegree	NA	NA	NA	1105.17	834.76	1216.61

To select the graph type, we calculated accuracy for genre prediction:

a) using single feature at a time.
b) using pair of features of all possible combinations

After calculating individual accuracy of single features, the average over all the accuracies is shown in Table 5 for three different classifiers KNN, SVM and Naïve Bayes. The experiment was then done using pair of features and Table 6 shows the results.

Since on average, non-directional graph data provided better accuracy with two out of three classifiers in the experiment when calculating average of accuracy of individual feature. Also, all the three classifiers provided better average accuracy with pair of features experiment. Hence, we decided to conduct rest of the analysis using non-directional graphs and since the Naïve Bayes and SVM provided almost identical accuracy with single feature accuracy and showed only a difference of 1% with pair of features we decided to conduct rest of the experiments with both the classifiers as we wanted to explore what features are considered better for genre prediction by each of these classifiers.

Table 5: Directional vs non-directional average of accuracies of all individual features using KNN, SVM and Naïve Bayes

	KNN	**Naïve Bayes**	**SVM**
Directional	44.87	51.85	49.14
Non-Directional	45.70	50.68	50.27

Table 6: Directional vs non-directional average of accuracies over all possible pair of features using KNN, SVM and Naïve Bayes

	KNN	**Naïve Bayes**	**SVM**
Directional	51.10	55.69	54.62
Non-Directional	53.57	56.38	55.63

4.3 Results

4.3.1 Single Feature Accuracy

Our first test attempt was to identify genre using only single feature at a time. However, no single feature was independently sufficient to identify the genre. As shown in Table 7, of the features tested, Path Length provided the greatest accuracy (66.43%) for genre identification with SVM. It is worth noting that this feature alone ties the accuracy produced with all features reported in the previous section. On the other hand, Total Number of Lines in the play was the best feature to identify genre using Naïve Bayes. However, both classifiers have graph density as the second-best feature for genre identification.

Table 7: Genre prediction using single feature.

SVM		**Naïve Bayes**	
Feature	**Accuracy**	**Feature**	**Accuracy**
Path Length	66.43	Lines	63.57
Graph Density	61.07	Graph Density	61.43
Diameter	58.57	Words	61.07
Characters	55.71	Path Length	60.71
Eigenvector	55.71	Average Weighted Degree	58.93
Eccentricity	55.71	Diameter	58.57
Harmonic	55.71	Connected Components	58.57
Average Weighted Degree	55.71	Characters	58.21
Lines	55.36	Eigenvector	58.21
Degree	55.36	Eccentricity	52.86
Closeness	52.50	Closeness	52.50
Connected Components	50.35	Modularity	50.00
Modularity	50.00	Degree	49.64
Words	47.50	Radius	47.14
Edges	47.14	Edges	46.78
Radius	47.14	Harmonic	44.64
Weighted Degree	44.28	Weighted Degree	43.93
Criticality	41.43	Criticality	41.07
Clustering Coefficient	38.93	Average Degree	38.21
Average Degree	33.21	Betweenness	35.71
Betweenness	27.85	Clustering Coefficient	22.50

4.3.2 Pair of Features Accuracy

When features were used in pairs, the network graphs achieved greater accuracy in identifying the genre of Shakespeare plays. Table 8 and 9 shows pair of metrics that were able to identify genre with accuracy higher than maximum individual feature accuracy for genre prediction.

Table 8: Pairs of features that provided above 70% accuracy in genre prediction using SVM classifier.

SVM		
Feature 1	**Feature 2**	**Accuracy**
Harmonic	Diameter	72.50
Harmonic	Path Length	72.50
Graph Density	Diameter	72.50
Graph Density	Path Length	72.50
Lines	Path Length	72.14

Table 9: Pairs of features which provided above 70% accuracy in genre prediction using Naïve Bayes classifier.

Naïve Bayes		
Feature 1	**Feature 2**	**Accuracy**
Lines	Eigenvector	77.86
Characters	Words	77.50
Words	Eigenvector	77.50
Words	Graph Density	75.00
Words	Path Length	75.00
Lines	Graph Density	75.00
Characters	Lines	74.64
Words	Eccentricity	74.64
Words	Diameter	72.14
Lines	Eccentricity	72.14
Lines	Diameter	72.14
Lines	Path Length	72.14
Lines	Modularity	72.14
Eigenvector	Diameter	72.14
Graph Density	Diameter	72.14

4.3.3 Multiple Features Accuracy

If we combine three features, the network graphs again achieve 10% higher accuracy in genre identification. Tables 10 and 11 show the triads that were able to identify genre with more than 80% accuracy. The best performance with SVM was with a set of Extracted Features (Words, Characters, Lines) at 83.57%. However, Words, Lines, and Eigenvector (a Node Feature) was essentially tied a 83.21%. With Naïve Bayes, a combination of an Extracted Feature (Lines) and two Graph Features (Graph Density and Degree) performed best.

Table 10: Sets of three features which provided above 80% accuracy in genre prediction using SVM classifier.

SVM			
Feature 1	**Feature 2**	**Feature 3**	**Accuracy**
Words	Characters	Lines	83.57
Words	Lines	Eigenvector	83.21
Words	Lines	Closeness	81.07
Lines	Eigenvector	Path Length	80.71
Lines	Harmonic	Path Length	80.71

Table 11: Sets of three features which provided above 80% accuracy in genre prediction using Naïve Bayes Classifier.

Naïve Bayes			
Feature 1	**Feature 2**	**Feature 3**	**Accuracy**
Lines	Graph Density	Degree	80.71
Characters	Words	Lines	80.36
Words	Criticality	Graph Density	80.36

Because of the exponential nature of exploring all combinations of all features, we did not do an exhaustive test of all combinations of 4, 5, 6, etc., features. However, we continued testing by adding additional features one by one to well-performing feature sets to see if we could further improve accuracy. After this exploration we found that, with Naïve Bayes, the feature set of

Characters, Words, Lines and Path Length provided 86.07% accuracy. Our highest accuracy was with the feature set of Words, Lines, Closeness, Graph Density and Average Weighted Degree provided 88.93% accuracy with SVM. This feature set captures a combination of Extracted, Node, and Graph features, indicating that all are important for accurate genre prediction.

4.4 Using the Genre Predictor on Disputed Play

4.4.1 Disputed Plays

To apply our findings in the literary world, we investigated the genre classification of Shakespeare's Roman, Romance, and Problem plays. Table 12 list these plays along with the most commonly attributed genre. However, the genre of these sets of plays is in some dispute among literary scholars and we felt that it would be interesting to see how our SNA predictor classified them.

Table 12: Disputed Plays

Category	**Play Name**	**Usual Genre**
Romances	The Tempest	Comedy
	The Winter's Tale	Comedy
	Pericles Prince of Tyre	Comedy
	Cymbeline	Tragedy
Roman	Antony and Cleopatra	Tragedy
	Coriolanus	Tragedy
	Julius Caesar	Tragedy
	Titus Andronicus	Tragedy
Problem	All's Well That Ends Well	Comedy
	Measure for Measure	Comedy
	Troilus and Cressida	Tragedy

4.4.2 Classification of Disputed Plays using SVM and Naïve Bayes

We classified each play using the best genre predictor features set with SVM and Naïve Bayes. Thus, we trained an SVM classifier using a feature set comprised of Words, Lines, Closeness, Graph Density and Average Weighted Degree and also a Naïve Bayes classifier trained using the Characters, Words, Lines, and Path Length features. For the Romances, we trained on 32, all plays except the 4 Romances, and then classified the Romances using that classifier. Similarly, for the Roman plays, we trained on all 32 non-Roman plays and classified the Roman plays. Finally, for the three Problem plays, we trained on the other 33 plays and then predicted the genre for the held-back 3 Problem plays.

Table 13 shows the accuracy of genre classification using best features of each classifier. In this cases, accuracy measures is how often our classifier predicted the most-commonly associated genre for the play, i.e., it agreed with the most common genre. As shown in Table 13, Naïve Bayes agreed with the Problem Plays' usual genre 100% of the time, but many of the other predicted classifications were different. The Romances, in particular, have social networks that only match their usually-associated genre 50% of the time regardless of which classifier is used.

Table 13: Disputed Plays Accuracy

Category	SVM Accuracy	Naïve Bayes Accuracy
Roman	75.00%	50.00%
Problem	66.67%	100.00%
Romances	50.00%	50.00%

Tables 14, 15, and 16 show our more detailed results that would be of interest to literary scholars. From Table 14, we can see that whereas both classifiers agree that *The Tempest* is a Comedy, *The Winter's Tale*, also usually considered a Comedy, looks like a History to both our classifiers.

Table 14: Original and predicted classes for Romances

Play Name	Original Genre	SVM	Naïve Bayes
The Tempest	Comedy	Comedy	Comedy
The Winter's Tale	Comedy	History	History
Pericles Prince of Tyre	Comedy	Tragedy	History
Cymbeline	Tragedy	Tragedy	Tragedy

From Table 15, we can see that whereas both classifiers agree that *Coriolanus* is a Tragedy, *Titus Andronicus*, also usually considered a Tragedy, looks like a Comedy to both our classifiers.

Table 15: Original and predicted classes for Roman plays

Play Name	**Original Genre**	**SVM**	**Naïve Bayes**
Antony and Cleopatra	Tragedy	Tragedy	History
Coriolanus	Tragedy	Tragedy	Tragedy
Julius Caesar	Tragedy	Tragedy	Tragedy
Titus Andronicus	Tragedy	Comedy	Comedy

From Table 16, we can see that whereas both classifiers agree that *All's Well that Ends Well and Measure for Measure* are both Comedies. However, *Troilus and Cressida*, also considered a Tragedy, looks like a Tragedy to our Naïve Bayes classifier but is predicted to be a Tragedy by our SVM classifier.

Table 16: Original and predicted classes for Problem plays

Play Name	**Original Genre**	**SVM**	**Naïve Bayes**
All's Well That Ends Well	Comedy	Comedy	Comedy
Measure for Measure	Comedy	Comedy	Comedy
Troilus and Cressida	Tragedy	Comedy	Tragedy

4.5 References

1) https://en.wikipedia.org/wiki/Centrality

2) https://gephi.org/developers/

3) https://weka.wikispaces.com

4) https://www.cs.waikato.ac.nz/~ml/index.html

5) https://www.csie.ntu.edu.tw/~cjlin/libsvm/

6) Shukla Manisha, Susan Gauch and Lawrence Evalyn. 2018. Theatrical Genre Prediction Using Social Network Metrics. In 10th International Conference on Knowledge Discovery and Information Retrieval, Seville, Spain. (Accepted).

5. FUTURE WORK

Since the parser is highly extensible and can be used with any plays encoded in TEI, future work applying these methods to literary analysis does not need to be restricted to plays that are similar to Shakespeare's but could be used to compare plays over a long period of time. Future work does not even need to be restricted to plays written in English; one future application in development, for example, will study eighteenth century plays written in English, French, and German. As we develop our website, we will add functionality for others to upload their own TEI encoded plays and download the resulting Gephi file, enabling broad applicability of our methods to new literary research problems.

Future refinements to the social network generator could make edges between nodes directional, to better capture imbalanced relationships between characters; this level of detail was not necessary to distinguish between Shakespeare's plays, but might be important for different identification tasks. Natural Language Processing (NLP) could also be integrated into the parser to more accurately identify the targets of speech, to capture instances where characters are on stage but cannot hear what is being said or are not being spoken to. These kinds of improvements would reduce "false positives" in the creation of edges between nodes, perhaps enabling better analysis of larger or more complicated groups of literary plays.

6. CONCLUSION

In this work, we successfully classify plays based on their genre without using the actual vocabulary of the plays. Our networks of the well-studied works of Shakespeare can provide a baseline against which to contextualize similar studies of other plays. The network graphs themselves provide a new insight into the plays, revealing the hidden shape of social relationships between characters. The application of mathematical graph analysis to these networks provides a dramatically faster and more scalable way to determine important information about them, in this case their genre.

We collect and parsed 36 TEI-encoded plays by William Shakespeare and parsed them to identify the lines and words spoken by each character to all other characters in that scene. We used this information to create a social network graph for each play in which each node was a character with edges representing the number of words and lines spoken between two characters over all the scenes in the play. In total, we represented each play using 21 features, 4 features extracted from the text, 8 features extracted from the nodes in the social network graph, and 9 features that summarized attributes of the resulting overall graph.

We first investigated several classifiers for our application. We found that Naïve Bayes and SVM classifiers outperformed KNN classifiers when predicting genre trained on all features. Based on this result, we used Naïve Bayes and SVM classifiers for all further experiments. Next, we examined the impact of directed vs undirected links between characters. The results indicated that undirected links were more accurate 55.63% vs 54.62%, so we used undirected links in our subsequent investigation.

Since the accuracy when trained on all features was quite low, 66.43%, we next investigated combinations of features that might give better accuracy. We began by looking at single features and found that the best single features were Path Length with 66.43% accuracy for SVM and Lines with 63.57% accuracy for Naïve Bayes. We then looked at feature pairs and found that the accuracy improved, with 20 different feature pairs producing an accuracy over 70%. We then investigated feature triples and found that 8 different feature triples produced accuracy over 80%. We observe that the sets of three factors that provide higher accuracy do not necessarily always include the features that were able to provide better accuracy as pairs. Many of the pairs, for example, include Graph Density or Path Length as one of the two identifying features, but none of the triples include graph density as a feature for maximizing the accuracy, and the triples instead include the number of words and lines as the most commonly useful feature. Instead, the triples include the Number of Words and Lines, two of the most accurate single features

Overall, the metrics seem to capture a specific kind of information about the play that is more effective in combination with other metrics. Total Number of Words, for example, is only able to provide 47.5% accuracy alone, but reaches almost 89% when combined with another Extracted Feature (Lines, two Graph Features (Closeness, and Graph Density), and a Node Feature (Average Weighted Degree).

To apply these findings to literary research, we have explored in more detail the genre attributions of Shakespeare's romances and problem plays (Evalyn, Gauch, and Shukla 2018).

7. REFERENCES

1) Evalyn, Lawrence, Susan Gauch, and Manisha Shukla. 2018. Analyzing Social Networks of XML Plays: Exploring Shakespeare's Genres. In Digital Humanities Conference 2018. https://dh2018.adho.org/en/analyzing-social-networks-of-xml-plays-exploring-shakespeares-genres/.