Large-scale Profiling of Movie Scenes and Character Types

FANG-CHIEH CHOU
Department of Biochemistry, Stanford University. fcchou@stanford.edu
and
JIN CHEN
Department of Applied Physics, Stanford University. jchen77@stanford.edu
and
YU-WEI LIN
Department of Electrical Engineering, Stanford University. yuweilin@stanford.edu

Past efforts in analyzing movies have been focused on recommending movies to users, but little has been done on profiling the movie and character types to find combinations or trends. Using the movie dataset, we have developed a method to profile characters from movie scenes. By applying text mining techniques to a dataset of nearly 10,000 movies with each scene annotated, we can build a movie timeline and movie and character features through 1-shingles and 2-shingles that capture the appearance and disappearance of objects, actions, and characters, as well as the relative occurrence frequencies of each item. We first created logistic classifiers to do binary classifications on the movie genres, which are labeled. Our logistic classifiers proved to be quite successful in classifying our movie features into the respective genres. We then applied the classifiers to our character features, mapping the characters onto each genre. From this, we were able to determine the character "types" based on the different genres and to analyze the frequent combinations of character types in each movie genre. Our method is easily applicable and extendable to study trending scenes and character combinations, in addition to performing recommendation based on actors and characters.

1. INTRODUCTION

Past efforts of machine learning and data mining in analyzing movies have been focused on recommending movies to users [4]. The algorithms mainly find similarities between users and movies for recommending movies, which focuses more on the user and very general categorizations of movies. The implementation of Latent factors, collaborative filtering, content-based filtering,...etc. have been proven to be quite successful in user-movie recommendations (for example, the Netflix challenge, [1]). However, little has been done on analysis and profiling of characters within movies, and comparing characters across different movies [5]. Analyzing characters and movie scenes are important, as it provides insights into the trending character types, trending scenes, and frequently appeared combinations of character types in movie genres. This could add an extra layer of information to the current recommendation systems that only focuses on users and movies.

Character profiling and personality analysis have traditionally been a field of study in psychology and social science, which focus on just one or several movies [7]. Character theory has been used to study the roles of the characters in each movie. However, these previous theories are based on the analysis of small datasets and on stereotypes [2]. It is therefore particularly useful to apply statistical learning methods to large movie datasets to understand the characters' profiles, which provides correlations and trends within characters that is more informative than just stereotypical analysis.

In this report, we analyzed a dataset of annotated movie scenes of nearly 10,000 movies to profile each character based on the actions, objects, and other characters that appear within each scene with them. From this analysis, we were able to classify not only the type of movie genres, but also the type of characters, in addition to trends and combinations of characters that often appear in a movie. This can be further applied to profile the types of scenes, movies, and characters that particular actors are good at or are better received. This type of profiling can be used for future recommendations of a new script to an actor, for recommending actors to directors, or for recommending movie types and character types for directors and script writers. Another aatural extension is making recommendations to users based on a specific plot, character, or actor. With a large amount of movie data available, including IMDB and Rotten Tomato reviews, ratings, and information, as well as the annotated movie scene information, it is now possible to mine this data to create a comprehensive profile of each character of the movies.

2. THE DATA

In order to profile movies and characters, we used the movie scenes dataset provided by Professor Jure Leskovec. The dataset contains 8,933 movies from 1918 to 2013, with every scene manually annotated. Each data file consists of an unique identifier for the movie and the movie metadata. The metadata consists of the movie name, year, a short description, the language, the director(s), the cast (which consists of actor names and character names), and the movie genres. In addition, there are Rotten Tomato identifiers and scores.

The bulk of the dataset, and the most useful information from the dataset, are the scene annotations. Every scene of each movie is marked with the start and end time of each segment, and annotated with (1) actions (for example, fighting, attacking, flying), (2) locations (for example garage, airport, gym), (3) objects (for example animal, drinks, books, drugs) and (4) appearances (for example, character, actor). There is one segment for each action, object, location, and appearance, and the segments can be overlapping, indicating the simultaneous appearance of the objects, characters, or actions. This can be analyzed to determine how long each object, character, or action last, as well as how they overlap in time.

The data is organized in a JSON format, so it can be easily parsed using Python.

3. RESULTS

3.1 Movie and character features

In order to analyze the movies and the characters, we must first create features that represent each movie or character to capture the appearances and correlations of the characters, objects, and actions.

These "movie features" or "character features" are matrices that summarize the characteristics of each movie or character.

To create the feature, we first generated a "timeline" that represents the entire duration of each movie. We separated the timeline into 30-second time bins. Then we filled each time bin with the scene segments that appear in it. For example, as shown in Figure 1, House and character 1 appeared in the first time bin. House and character 1 continued to appear in the second time bin, in addition to Gun and character 2. We repeat this for the entire duration of each movie, so we can obtain a discretized timeline that captures the appearance, disappearance, and co-occurrence of objects, characters, and actions.

To build movie features from our timeline, we took the concept of w-shingling in text mining [3]. We created three types of shingles for each timeline: 1-shingle, 2-shingle vertical, and 2-shingle horizontal. 1-shingle is just a histogram of counts for all the items that appear in each movie. For example, in Figure 1, House appeared 3 times, Gun appeared once, character 1 appeared twice, and so on. 2-shingle vertical counts all the doublets of items that appear within each time bin, which captures the co-occurrence of items. For example, the first time bin has the combination (House, 1); the second time bin has the combinations (House, Gun), (House, 1), (House, 2), (Gun, 1)... etc. Note that the order of items in these shingles does not matter here. Similarly, we built a histogram of counts for all these 2-shingles as our feature. 2-shingle horizontal is similar to 2-shingle vertical, but we counted the doublets of items that appear in two consecutive time bins. For example, the first two time bins have the combinations (House, House), (House, Gun), (House, 1), (House, 2), (1, House)... etc. Again, we constructed a histogram of counts for all these 2-shingles. This feature captures the appearance and disappearance of items in consecutive time bins. It is important to note that the order of these shingles does matter. The three types of histogram features were then concatenated to create the final feature used in this work.

The character features were created in a similar fashion. Instead of counting all possible shingles in each movie, we just used the time bins that the target character appears. For example, character 1 shows up in time bins 1 and 2, but not 3. So for the feature of character 1, we just used 1-shingles and 2-shingles in the first two time bins.

Finally, to normalize the importance of each shingle properly, we applied TF-IDF (term frequencyinverse document frequency) [6], a commonly used normalization scheme in text mining, to our feature matrix. The feature for each movie/character is further normalized to have a unit norm.

3.2 Initial attempts in character classification

Using the character features created using the method above, we first attempted to classify characters into different categories. However, a major difficulty is that the characters are not labeled, making it impossible for supervised learning. Unsupervised clustering is also difficult since we did not have a proper distance metric that would work for our features, which have 20,000 dimensions. In addition, it is quite difficult to interpret the meaning of each cluster in such a high-dimensional space. We have tested a few common distance metrics (Euclidean distance, cosine distance, etc.), but without much success.

We have also tried to manually label a few characters and classified characters in this small labeled dataset. For example, we manually labeled a few "superheros" such as Batman, Iron Man, Superman, Spider man, etc., and classified the characters as superhero or non-superhero by logistic regression, SVM, or random forest classifier. However, these classifiers performed poorly when applied to the full dataset. For example, Batman and the Joker were both classified as superhero, although the Joker is actually a vallain. Due to the limited training data, the classifier could only give a crude picture of the characters. For example, the previous Batman and Joker case can be explained intuitively since both batman and joker are "action"-type characters, and the classifier was probably just classifying based on "action" rather than "superheros". An additional complexity of these methods is that since most characters are from movies not widely watched, it is difficult to manually label the bulk of characters in our dataset.

Table I. Summary of Genre Classification Statistics

Genre	Accuracy	Precision	Recall	Number of movies
Drama	0.73	0.41	0.51	4817
Comedy	0.76	0.40	0.50	4320
Thriller	0.79	0.34	0.49	3121
Action	0.88	0.48	0.63	2401
Horror	0.88	0.44	0.52	2167
Comedy drama	0.84	0.16	0.28	1672
Adventure	0.90	0.32	0.43	1440
Science fiction	0.92	0.33	0.44	1133
Crime drama	0.89	0.20	0.39	1130
Documentary	0.97	0.66	0.85	24
Romantic comedy	0.91	0.23	0.38	1078
Fantasy	0.92	0.23	0.37	980
Romance	0.92	0.12	0.20	764
Western	0.99	0.75	0.84	444
Historical drama	0.96	0.14	0.2	456
War	0.96	0.23	0.44	369
Biography	0.95	0.13	0.26	370
Animated	0.97	0.28	0.46	250
Musical	0.96	0.23	0.53	356
Mystery	0.96	0.03	0.04	335
Children	0.97	0.16	0.38	153
Docudrama	0.97	0.04	0.08	204
Martial arts	0.98	0.35	0.60	167
Musical comedy	0.98	0.06	0.10	169
Dark comedy	0.99	0	0	81
Music	0.99	0.16	0.5	42

3.3 Movie genre classification

Due to the failure in our initial attempt on character profiling, we took a step back to work on movie genre classification, since movie genres are already labeled in the dataset. These movie genre classifications were then used to help classifying the characters, as discussed in sections below. We took the movie features, with each movie labeled with its genres (note that movies may have multiple genres), to do supervised classification. For each genre, we did a binary classification using logistic regression classifier, to predict if a movie is in the target genre. We also tested SVM and random forest classifiers; the results were similar and worse than logistic regression. For each genre, we trained the classifier with 90% of the data, and tested the performance with the remaining 10% of data. To prevent imbalance between number of positive (movies in target genre) and negative training examples, we oversampled the target genre movies in the training set to ensure a 1:1 ratio between positive and negative data points in our training set. We then apply the classifier to the test set to determine the accuracy, precision, and recall of the classifier, as summarized in Table I.

For each genre, we found that the accuracy is usually quite high, with a good precision and a good recall, especially for the top few

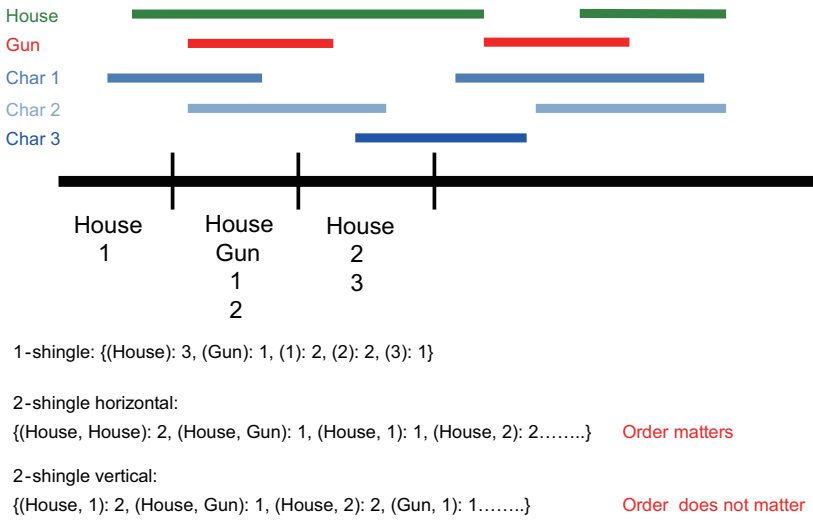

Fig. 1. Example movie feature and timeline. The timeline is separated into 30 second time bins, and the scene segments that appear in each time bin is counted. Then, 1-shingles, 2-shingle horizontal, and 2-shingle vertical features can be constructed from the timeline.

Table II. Selected examples of features

Genre	Selected Positive Features
Action	(Attacking, Jumping, vertical)
	(Running, Riding, horizontal)
	(Car, Hotel/motel/inn, horizontal)
	(Exploding, Ship, horizontal)
	(School, Weapon, vertical)
Drama	(Street/Road, Bedroom, horizontal)
	(Dining Room, Field, horizontal)
	(Magazines, Bags, horizontal)
	(House, Porch, vertical)
	(House, Smoking, vertical)

genres (which appears more often and is more critical). For the bottom genres, due to the limited number of examples in the data, the recall and precision were quite low. Furthermore, Western, a unique genre in our data, had very high accuracy, precision, and recall ($>$0.7) at the same time. For other more common genres, e.g. action, horror, and drama, the accuracies were limited to around 0.7 to 0.9 with precisions and recalls around 0.5. This is due to many movies of these types often cross multiple genres.

To understand what shingles are signatures for each genre classifier, we extracted the important feature components for each genre classifier, as shown in Table II. The features make sense. For example, for Action movies, the critical features involve combinations between actions like attacking, jumping, exploding, and objects like weapon and car. Drama movies, on the other hand, have important features involving bedroom, dining room, house, porch, etc.

3.4 Attempts to improve the movie and character features

Since now we have a good benchmark, we can try to optimize our features. It is possible that not only the items that appear in each time bin that is important, but the difference of items between each consecutive time bins can also important. We incorporated these "difference features" in addition to the 1-shingle and 2-shingles described above and ran feature selection with the genre classifier. However, the addition of these difference features did not significantly improve the performance of the genre classifier.

Furthermore, to test the importance of different single types, we also repeated the experiment with only 1-shingle, and the accuracy is general decreased around 0.2. Thus, this indicates the addition of the 2-shingles are crucial.

3.5 Analyzing characters

We proceeded with our original movie and character features without the incorporation of the difference features and feature selections to keep things simple. With our previous results in movie classification, we applied the logistic regression classifiers for each genre, trained with the movie features, to each character feature. We then calculated the probability that each character exhibits a particular genre type. Essentially, we are mapping the character feature onto each movie genre. One example is shown in Table III, which analyzed Batman and Joker from the movie The Dark Knight (2008). As expected, Batman exhibits high scores in Thriller (0.96), Action (0.37), Adventure (0.97) and Crime Drama (0.74). Batman also has a high score in Science Fiction (0.99), probably due to his extraordinary powers and technology. As portrayed throughout the movie, Batman has a high level of mystery, as indicated by his high score in Mystery (0.87). Thus, the analysis of Batman makes intuitive sense. If we compare the Joker from the same movie, we can see that many of the scores are similar. Though, it is interesting to note that while Batman has a very low Horror score, the Joker has a very high score (0.96). Moreover, Batman has a very high score in Mystery (0.87), while the Joker has a relatively low score (0.002). Thus, our character profiling method is not just picking out the genres of the movies where the characters belong to; instead, we are able to separate out characters of different types within the same movie.

Table III. Example Character Analysis of the Dark Knight

Genre	Dark Knight/Batman	Dark Knight/Joker
Drama	0.005083	0.001103
Comedy	0.376786	0.034829
Thriller	0.961771	0.991281
Action	0.365348	0.992663
Horror	0.001306	0.959315
Comedy drama	0.036339	0.000007
Adventure	0.971452	0.788594
Science fiction	0.997813	0.372594
Crime drama	0.744358	0.001417
Documentary	0.000039	0.000000
Romantic comedy	0.000000	0.000000
Fantasy	0.385773	0.984121
Romance	0.001062	0.000100
Western	0.000182	0.000550
Historical drama	0.000002	0.003090
War	0.000016	0.000000
Biography	0.000423	0.000005
Animated	0.062279	0.001833
Musical	0.000011	0.000045
Mystery	0.868754	0.002359
Children	0.000138	0.000002
Docudrama	0.000628	0.000000
Martial arts	0.002699	0.031714
Musical comedy	0.000004	0.000000
Dark comedy	0.009716	0.000000
Music	0.000000	0.000145

3.6 Validation: statistical analysis and trending of character types

Though manually picking out interesting characters, the character profile seems to make intuitive sense. However, we need a method to statistically validate our method and results, using the entire dataset instead of just observing individual examples. In following paragraphs, we will describe two comprehensive analysis of all characters in our dataset, by analyzing the distribution of types of characters in each movie genre, and by analyzing the frequency of combinations of character types in different movie genres.

To gain more insight into the results, we plotted the histogram of the probability scores for each character types for different movie genres, as shown in Figure 2. The figure shows that in a certain movie genre, the distributions of genre types are different. For example, in Horror movies, the characters typically have high Thriller scores and low Action scores. Moreover, we can see that the distribution of types varies with the movie genres. For instance, characters in Drama movies have a quite uniform distribution in Drama type; on the other hand, the characters in other genres, such as Action and Horror, tends to be less "Dramatic", with distributions peaking near zero. It is important to note that the y-axis is log-scale. Furthermore, there are two clear peaks in some distributions, indicating that in the some movie genre, many characters have high tendency to a certain type while many others have low tendency. Take Action movies for example, some characters seem to be very "Horror" but some are very "non Horror", and this might be a useful feature for separating the characters.

Second, we analyzed the character dataset to see if any particular combination of character types in the same movie appears much more frequently in each movie genre. For each character, we selected the top three genres with highest probability scores as the "characteristic types" of the character. The genres of the movie that the character is from were excluded, as it is used in training our character genre classifier. For each pair of characters in the same movie, we then extract all pairs of character types. For example, suppose we have a character Alice, with types Drama, Fantasy and Thriller, and another character Bob in the same movie classified as Action, Horror and Thriller, we then have character type pairs such as (Drama, Action), (Fantasy, Action), (Drama, Horror), etc. We first generated the baseline by computing histogram of character genre pairs for all possible character pairs in our dataset. For each movie genre, we counted the number of character type pairs for every pair of characters in the same movie, to see if different movie genres will give different distributions of commonly appeared character genre pairs. As shown in Fig. 3, indeed many combinations of character types appear more often than the baseline values for different movie genres. Table IV summarizes the top combinations appear in different genres. For example, Action genre movies tend to have character combinations of (Horror, Science Fiction), (Science Fiction, Crime Drama)...etc. Drama movies, on the other hand, tend to have combinations of (Comedy drama, Romance), (Comedy drama, Biography),...etc. Due to space limitations, we only showed a few examples, but this analysis works across all the different genre types. Thus, different movie genres tend to have different character type combinations, as expected. In Table V, we showed a few movie examples with characters of different types in the same movie. We have the example described before: Batman from the Dark Knight is a Science Fiction (or Crime Drama or Mystery), while the Joker is mainly Horror. From the result, we found that our character type assignments give statistically significant difference to the baseline across all the movie data. The obtained character type combinations also gave additional insights on the combination of character types preferred in different movie genres.

3.7 Impediments and difficulties

The main difficulty in doing character analysis is the lack of labeled data, in addition to limited knowledge of sufficient number of characters to do manual labeling. Thus, it is difficult to do a straightforward unsupervised clustering or supervised learning. Furthermore, there is usually a very broad spectrum of character types, so even if we have an appropriate distance metric, we may not be able to cluster the characters in a very meaningful way. Another difficulty is the lack of detailed scene annotations. Even though each scene is annotated, the tags are quite general (for example, car, explosion, vehicle, house...etc). If the tags can be more detailed, more insight can probably be mined from the data. Finally, although we were able to use the co-occurrence, appearance and disappearance of items to capture their correlation, the exact association between items can be ambiguous. For example, suppose we have two characters Alice and Bob and a gun appeared together in a scene, it may represent various scenes, such as (1) Alice shooting Bob with the gun, (2) Bob shooting Alice with the gun, (3) Bob holding a gun to guard Alice, etc. Since these scenarios lead to very different interpretations of the types of the characters, it is challenging to correctly profile each character with the data.

4. FUTURE WORK

Although we have developed a method to do large-scale profiling for character analysis, more work can be done. The movie and character features can be further improved by including additional information, for example the actor(s), year, director...etc for each movie or character. Furthermore, the scene annotations actu-

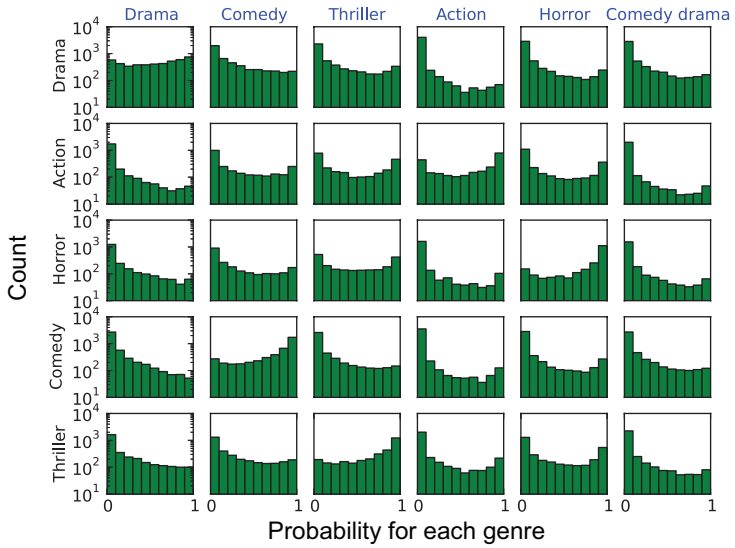

Fig. 2. Histogram of character genre feature in different movie genres. The figure only shows the histograms of the first 6 types in 5 different movie genres. The x-axis is the probability from 0 to 1, divided into 10 bins in the histograms, and the y-axis plots the counts of each bin in log scale.

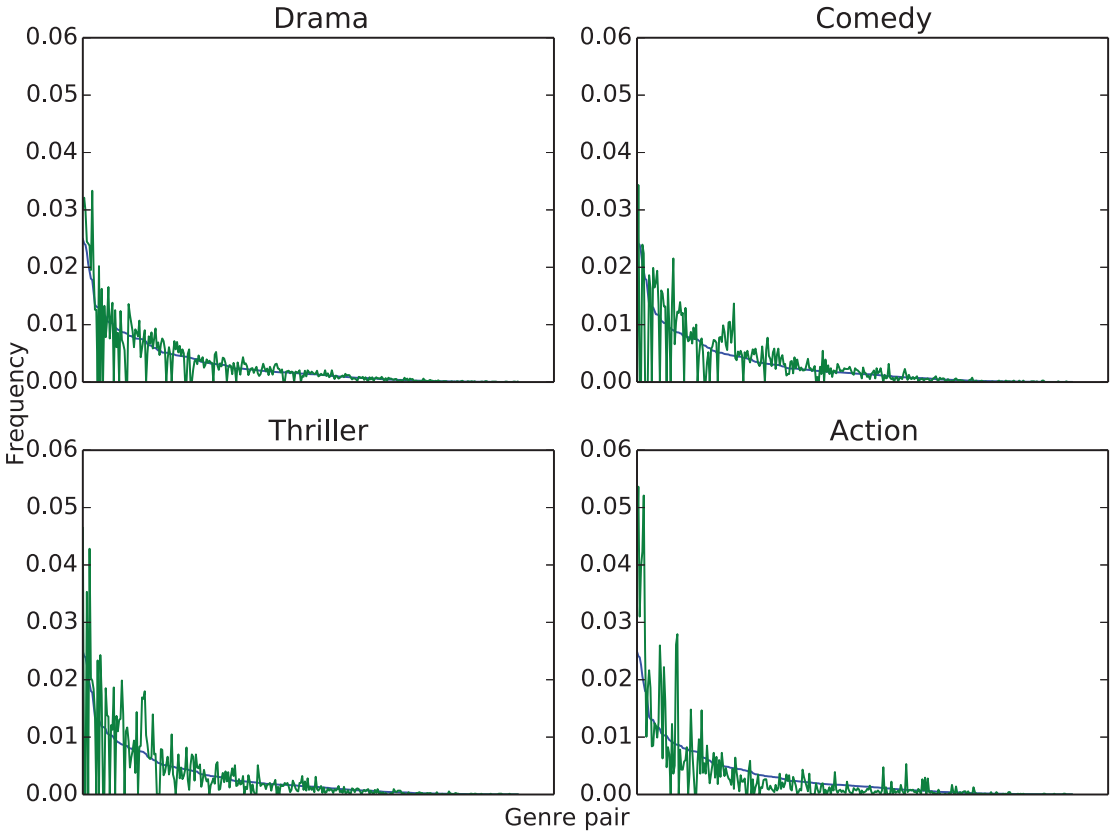

Fig. 3. Histogram of genre pairs for characters from four selected movie genres. The x-axis plots the genre pairs, with each pair indexed as an integer (not shown). Also the genre pairs are sorted based on their probability in the baseline (blue). The y-axis plots the frequency of each genre pair. The green line plots the histogram for the genre pairs from the particular genre. The blue line plots the base line when movies are randomly selected from any genre. Since the x-axis is sorted based on the blue line, the blue line monotonically decreases.

Table IV. Character genre pairs with highest probability compare to the baseline in different movie genres.

Drama	Comedy	Thriller	Action
(Comedy drama, Biography)	(Comedy drama, Romantic comedy)	(Crime drama, Mystery)	(Horror, Science fiction)
(Comedy, Comedy drama)	(Drama, Comedy drama)	(Horror, Mystery)	(Science fiction, Crime drama)
(Comedy drama, Romance)	(Drama, Romantic comedy)	(Drama, Mystery)	(Horror, Crime drama)
(Comedy, Romantic comedy)	(Comedy drama, Children)	(Action, Crime drama)	(Thriller, Science fiction)
(Comedy drama, Musical)	(Fantasy, Musical)	(Comedy, Mystery)	(Thriller, Crime drama)
(Comedy drama, Fantasy)	(Fantasy, Children)	(Action, Science fiction)	(Comedy, Science fiction)
(Fantasy, Biography)	(Adventure, Children)	(Science fiction, Crime drama)	(Adventure, Science fiction)

Table V. Examples of character combinations in movies

Movie	Character (genre)	Character (genre)
Pirates of the Caribbean: Dead Man's Chest	Capt. Jack Sparrow (Horror)	Elizabeth Swann (Musical)
The Dark Knight	Bruce Wayne/Batman (Science Fiction)	The Joker (Horror)
Spider-Man 2	Spiderman (Science Fiction)	Harry Osborn (Horror)
Sherlock Holmes: A Game of Shadows	Sherlock Holmes (Science Fiction)	Dr. John Watson (Comedy)
Thor	Jane Foster (Romance)	Odin (Horror)
RoboCop	RoboCop (Thriller)	Dick Jones (Crime Drama)

ally contain "captions", which include some of the lines spoken in each movie. However, since the captions are not labeled with which character actually spoke the line, and analyzing lines require natural language processing, it is beyond what we can accomplish in a quarter.

Currently, we only focused on the analysis of characters with respect to each genre type. It is possible to combine this analysis with IMDB scores and Rotten Tomato scores to analyze the popular and most well-received character types and character combinations within a movie. Furthermore, adding the actor information would allow us to profile the type of scenes, movies, and characters that a particular actor is good at or is the best received (as mentioned in the Introduction). This type of profiling can be used for future recommendations of a new script to an actor, or for recommending actors to directors, or for recommending movie types and character types for directors and script writers. Natural extension of this is make recommendations to users based on a specific plot, character, or actor.

Other ideas that are easily extended with our method, but is not directly related to character analysis, is to analyze trending scenes. Currently, we analyzed all the movies together. However, we can parse the movies with respect to their year, and try to find correlations between the years to determine if there are any trending scenes or character combinations that have become popular in recent years.

5. CONCLUSION

In conclusion, we have developed a method to profile characters from movie scenes. By applying text mining techniques to a dataset of nearly 10,000 movies with each scene annotated, we can build a movie timeline and movie and character features through 1-shingles and 2-shingles that capture the appearance and disappearance of objects, actions, and characters, as well as the relative occurrence frequencies of each item. Although a direct analysis on the character features proved to be difficult due to unlabeled characters, we first created a logistic classifier to do binary classifications on the movie genres, which is labeled. Our logistic classifier proved to be quite successful in classifying our movie features into the respective genres. We then applied this to classify our character features and map the characters onto each genre. From this, we were able to determine the character "types" based on the different genres and analyze the frequent combinations of characters in each movie genre. Our method is easily applicable and extendable to study trending scenes and character combinations, in addition to performing recommendation based on actors and characters.

ACKNOWLEDGMENTS

We want to thank Professor Jeffrey Ullman for his continuous guidance and support. We also would like to thank the CS341 staff, Jure Leskovec, Anand Rajaraman, and Rok Sosic for organizing this course.

REFERENCES

BELL, R. M., AND KOREN, Y. Lessons from the netflix prize challenge. *SIGKDD Explor. Newsl. 9*, 2 (Dec. 2007), 75–79.

HINTON, P. R. *Stereotypes, cognition and culture*. Psychology Press, 2013.

MANBER, U. Finding similar files in a large file system. In *Proceedings of the USENIX Winter 1994 Technical Conference on USENIX Winter 1994 Technical Conference* (Berkeley, CA, USA, 1994), WTEC'94, USENIX Association, pp. 2–2.

MELVILLE, P., MOONEY, R. J., AND NAGARAJAN, R. Content-boosted collaborative filtering for improved recommendations. In *AAAI/IAAI* (2002), pp. 187–192.

PARK, S.-B., OH, K.-J., AND JO, G.-S. Social network analysis in a movie using character-net. *Multimedia Tools and Applications 59*, 2 (2012), 601–627.

RAJARAMAN, ANAND, J. D. U. Cambridge University Press, 2011.

WEDDING, D., AND BOYD, M. *Movies & Mental Illness: Using Films to Understand Psychopathology*. McGraw-Hill, 1999.