# Script Sentiment-Based Movie Recommendation

Jonny Salama, Tanzila Morshed, Ryan Gibbons

# INTRODUCTION AND MOTIVATION

- Movie recommendation systems: content-based and collaborative filtering
- Content-based systems use simple attributes like **genre**, **cast**, **director**, etc.

- Motivation
  - Better recommend movies on more specific aspects of the script (plot aspects, dialogue and emotions)

# DATA AND RESOURCES

- Scripts from IMSDB database (until 2015)
  - 1,200 scripts, organized by genre
- Words by Movie Genre dataset (Kaggle)
  - 42,625 words

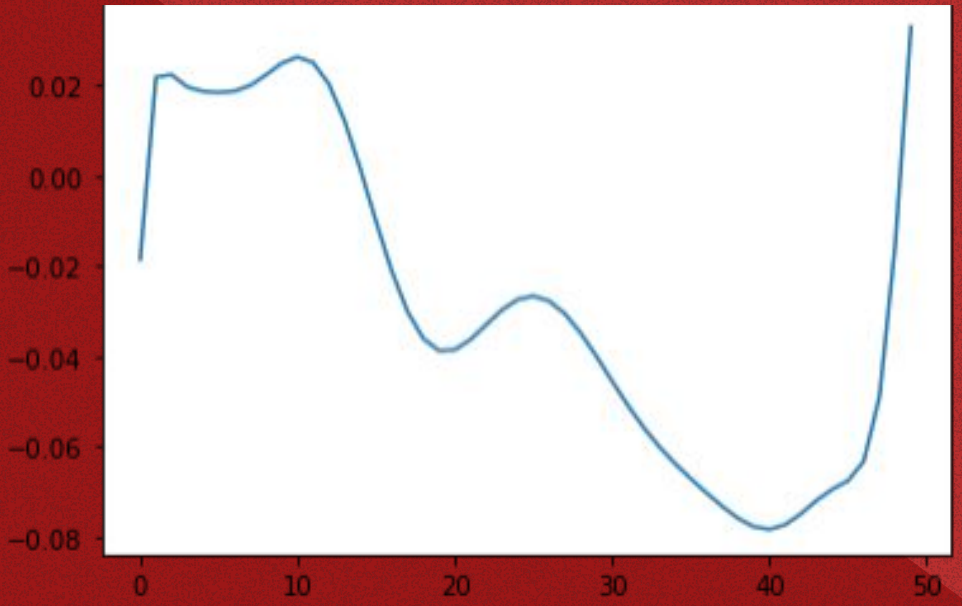
| text | target |
| --- | --- |
| Ex: inefficient | action |

# METHODS

- Script Analysis
  - TextBlob and VADER for sentiment analysis of scripts
- Genre clustering
  - Creating feature vectors from genre keyword frequency
  - k-means clustering
- Recommender
  - Cosine similarity matrix
  - BERT word embeddings

# EXPERIMENTS: SCRIPT SENTIMENT

- Goal: generate sentiment analysis graph for plot of film script



# METHOD

- Break script into individual ***scenes*** to analyze sentiment of each scene
- Calculate sentiment score for dialogue, descriptions, and cues  in the scene and produce weighted average sentiment for the scene
- Graph sentiment scores for each scene across the script, and fit curves to the series of scenes

# FEATURES

**Scene Length**
- Number of lines
- Dialogue vs. Description
- End of Scene

**Characters**
- Character count in script
- Dialogue frequency

**Subjectivity**
- Score from 0.0 to 1.0
- Dialogue vs. Description

**Cues**
- Provide implicit and expressive information

```
Characters: [('RYAN', 0.6584587323301413, 266), ('JESSICA',
0.24122207022343817, 161), ('GREER', 0.04691178797100235, 71), ('MOONEY',
0.026140688833673, 53), ('CRAIG', 0.006784108992434183, 27), ('CHLOE',
0.0033594833282150067, 19)]
```

# WEIGHTING AND SMOOTHING

- How to determine scene scores for complex scenes?
  - Some scenes can have a large inner disparity in sentiments

```
Next Mrs. Jacobi -- WARM, SMILING -- watches Donald ride the
bike. Now Graham plays the Leeds' tape, too. The dog runs to
camera… His tail wagging, his tongue out; he's a friendly
dog. Mrs. Leeds enters and pets him...
Her face is bruised and swollen from where Dollarhyde hit
her. She feels around on the bed. She's very FRIGHTENED.
She's trying to control herself...
```

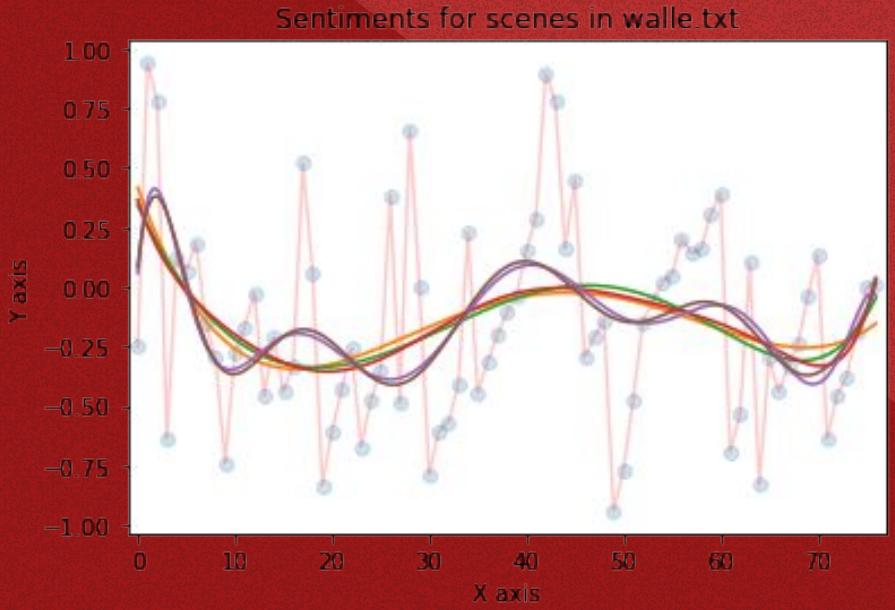
# WEIGHTING AND SMOOTHING

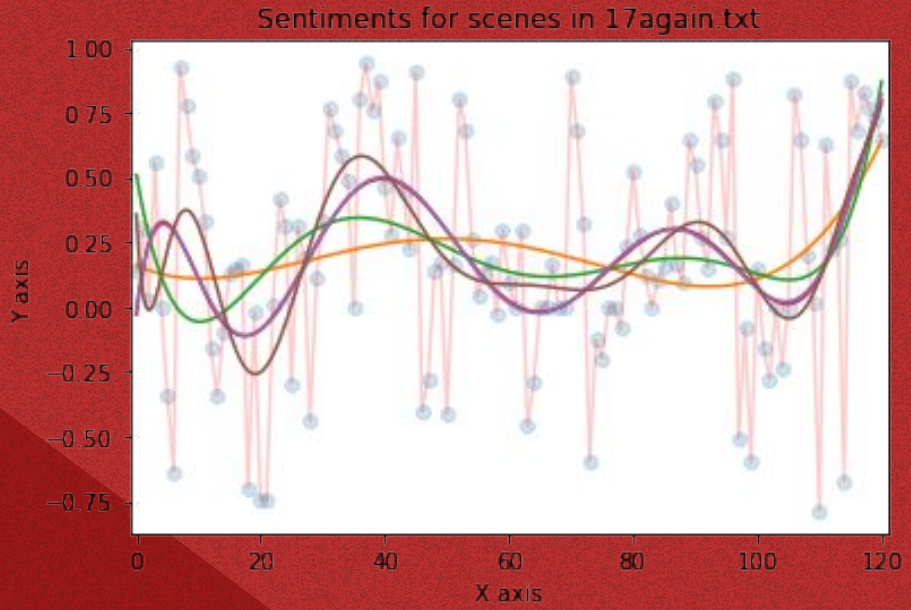
- Solution: Chunking
  - Split scene into chunks of max size = 10 sentences
  - Calculate sentiment score of each chunk
  - Distribution of chunk scores from -1.0 to 1.0
    - If distribution is even -> no weighting
    - If uneven (opposing spikes at beginning and end)
      - -> weight end of scene and main character speech more heavily
    - If uneven (spikes between chunks)
      - -> weight mode score highest
      - -> weight end of scene and main character speech slightly more heavily

# WEIGHTING AND SMOOTHING

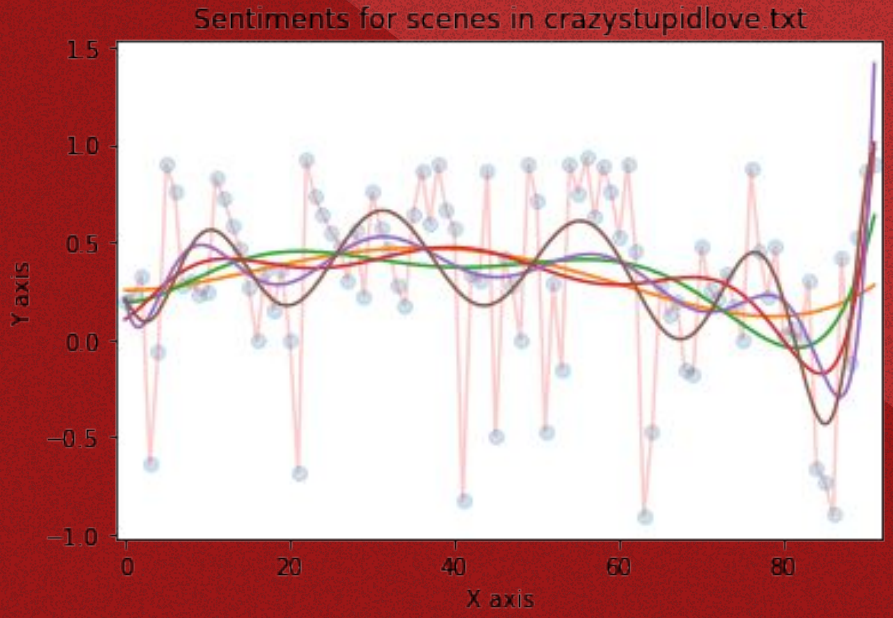
- Sometimes neighboring ***scenes*** have large disparity in sentiment scores, despite chunking algorithm
- Smooth scores using average sentiment score of **seen scenes**, bringing neighbors closer to the film average

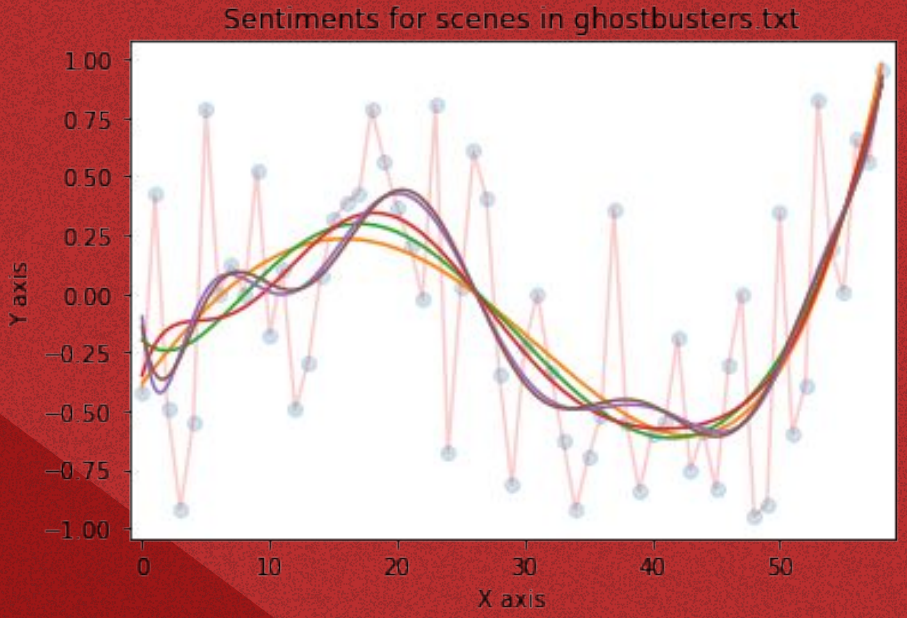
# RESULTS: SCRIPT ANALYSIS



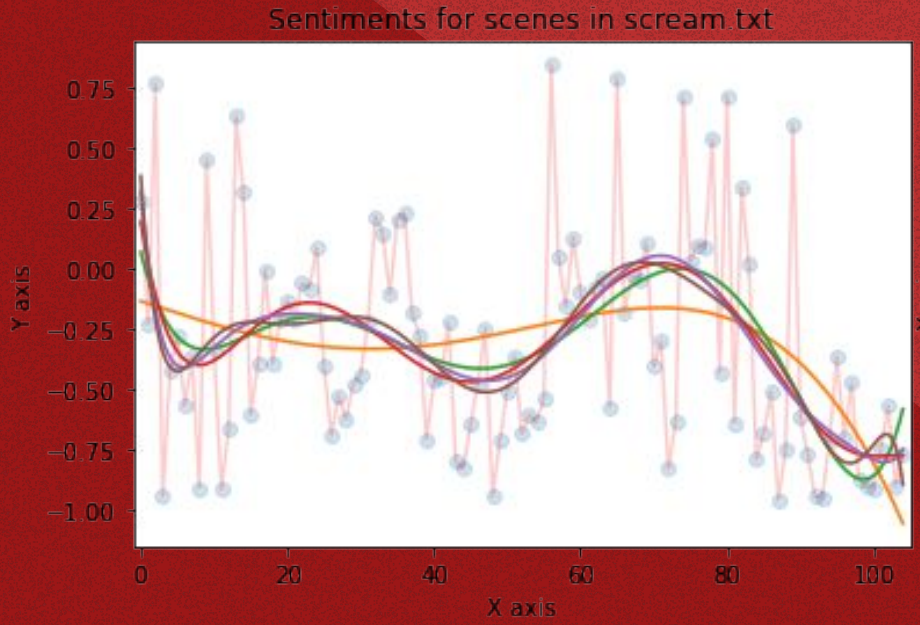



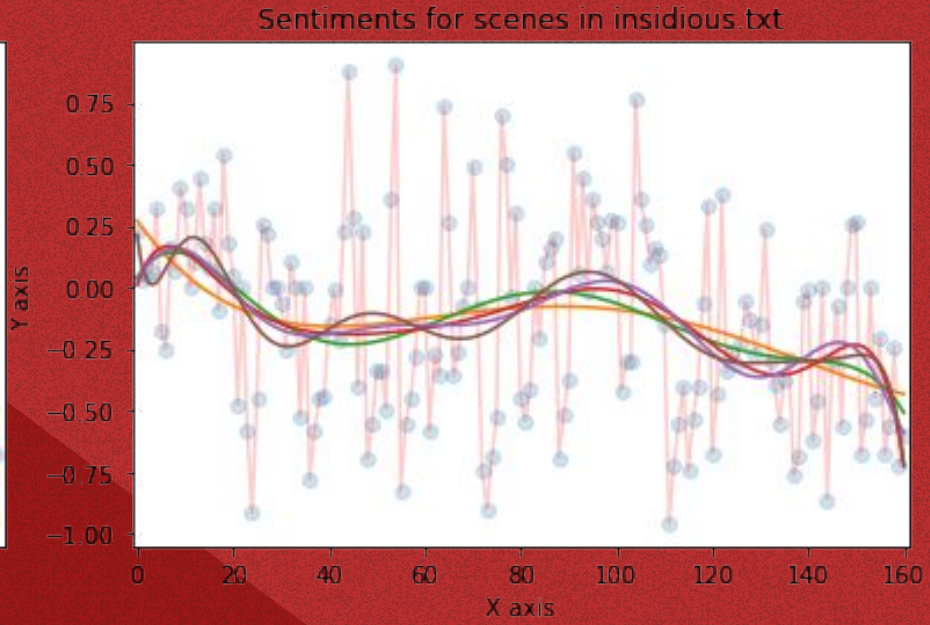
# RESULTS: SCRIPT ANALYSIS



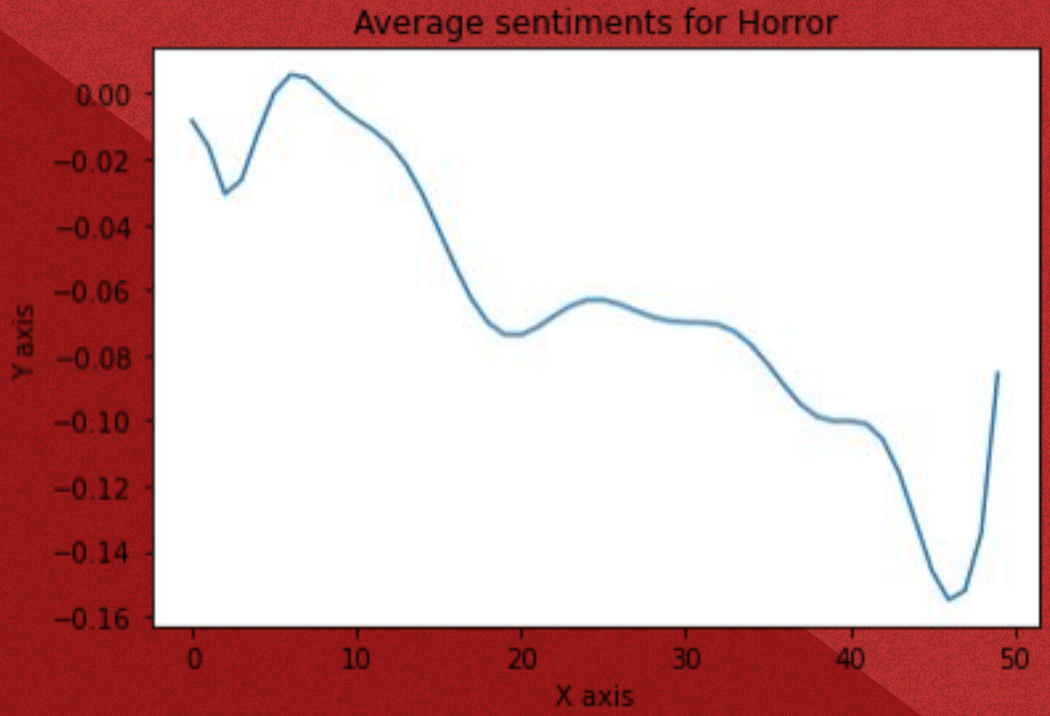



# RESULTS: SCRIPT ANALYSIS

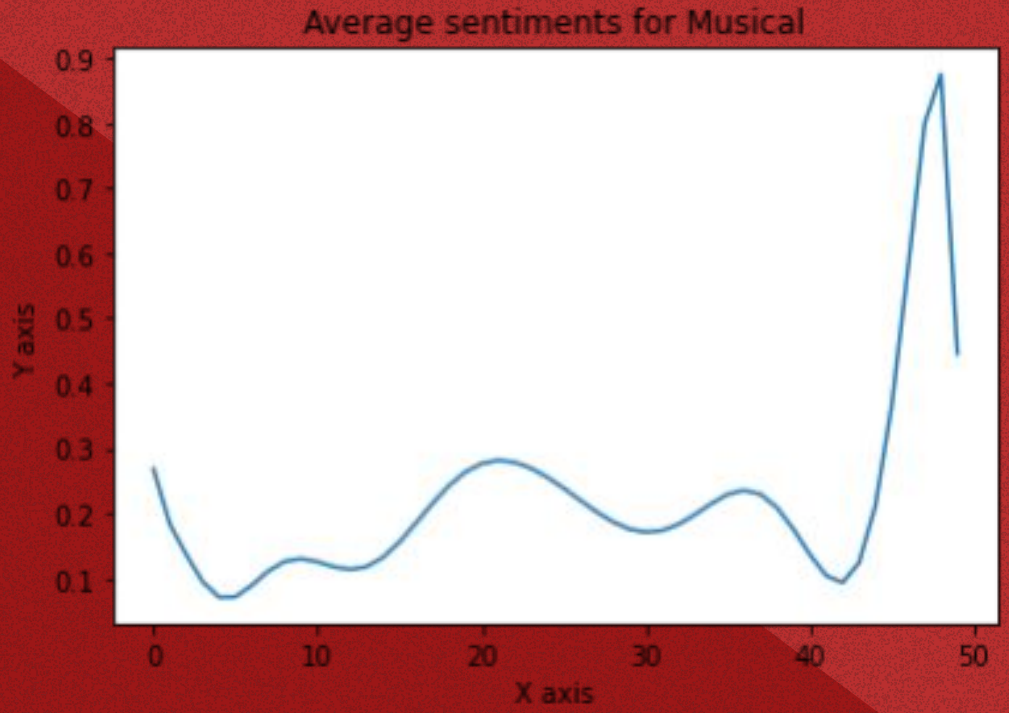


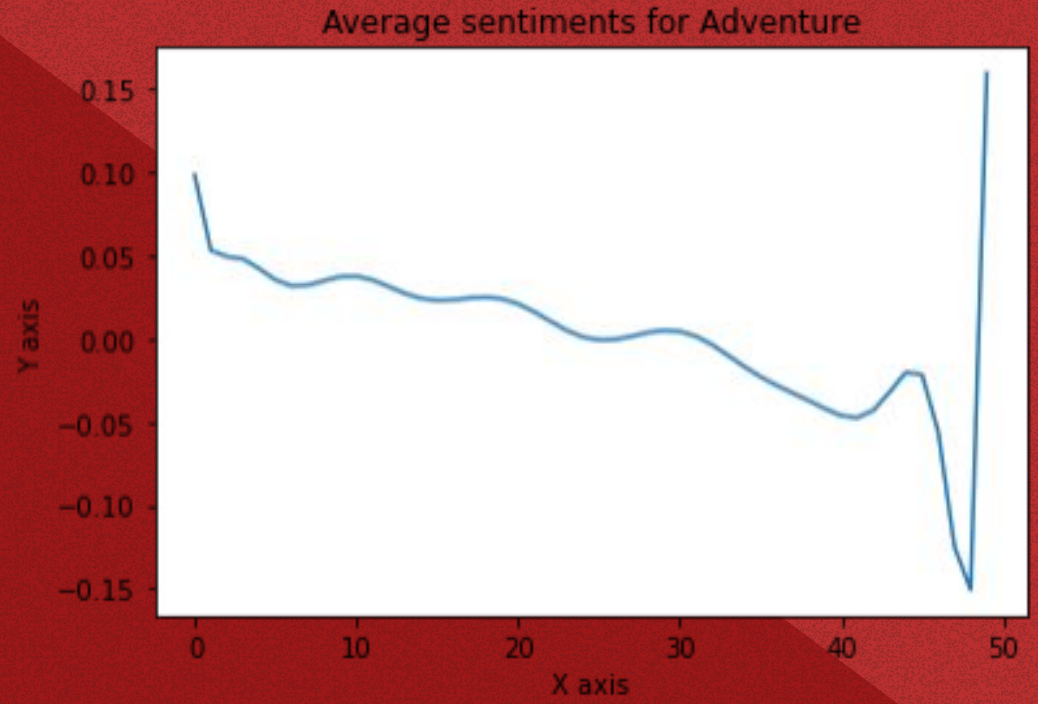
# RESULTS: SCRIPT ANALYSIS



# RESULTS: SCRIPT ANALYSIS



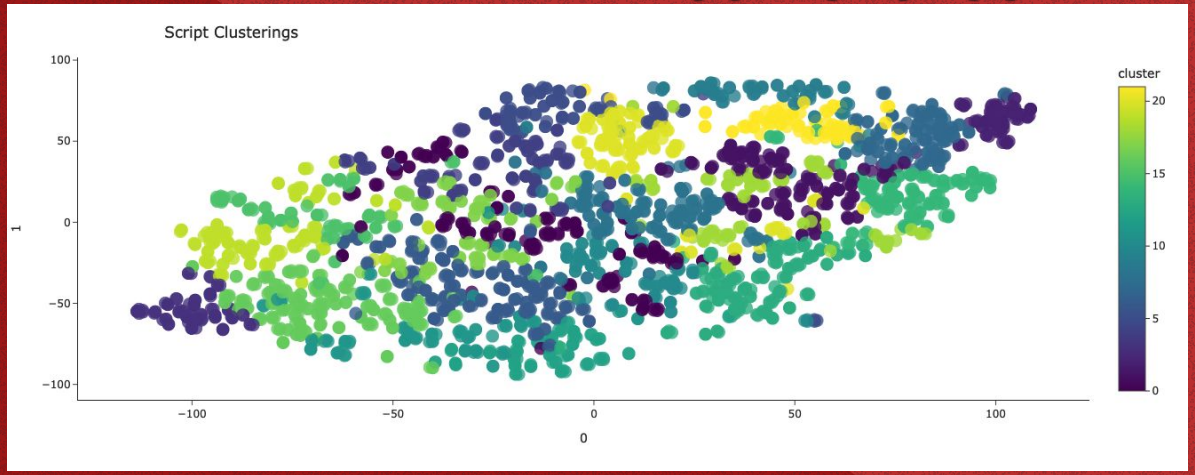
# RESULTS: SCRIPT ANALYSIS

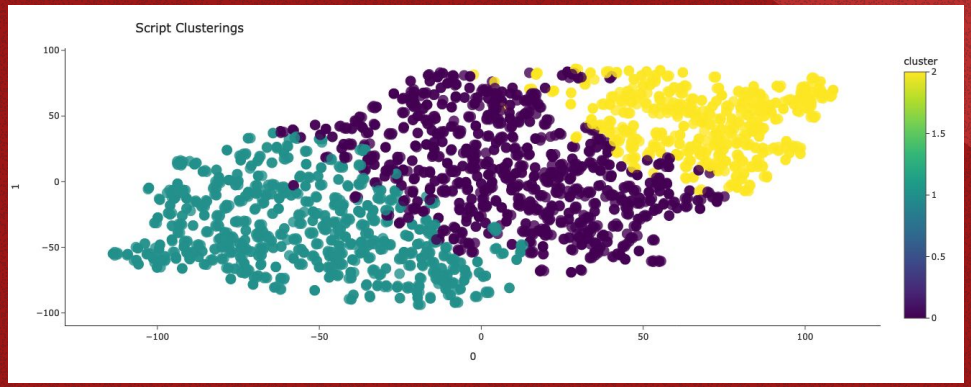


# EXPERIMENTS: CLUSTERING

- Movie script clustering by common keywords associated with different genres
- Clustering into 22 assigned genres as well as 3 genre groupings as described by Sureja (2016) and Thet et al. (2010)
- Rand Index & Adjusted Rand Index to measure quality of clustering

# RESULTS: CLUSTERING



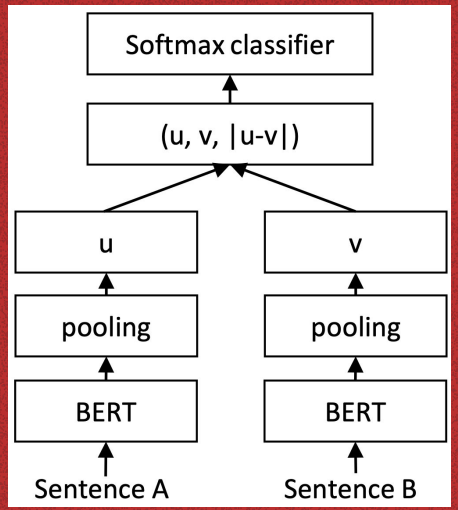

Adjusted Rand Index (ARI): Compares a predicted clustering to a set of predetermined labels, adjusted to weigh against the chance of randomness

ARI = .016

ARI = .083

# EXPERIMENTS: RECOMMENDER

- 3 features
  - Sentiment graph of a movie (weighted most)
  - Genre cluster vectors
  - BERT sentence embeddings of climax
- Cosine similarity matrix for each feature



# COSINE SIMILARITY MATRIX

- For every **movie i, row i** in a feature similarity matrix represents t**he list of similarity scores for all the movie vectors** with respect to the vector for movie i
- Produce lists of similarity scores for each feature
- Score movies with weighted average of each feature

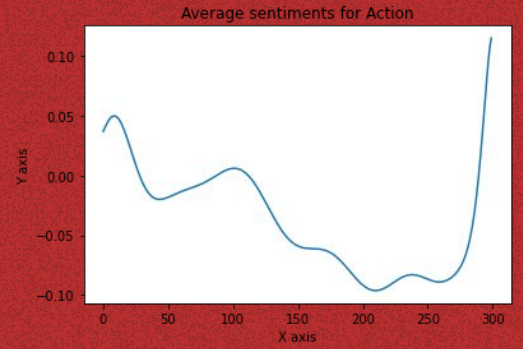
$$\text{similarity} = \cos(\theta) = \frac{\mathbf{A} \cdot \mathbf{B}}{\|\mathbf{A}\|\|\mathbf{B}\|} = \frac{\sum_{i=1}^{n} A_i B_i}{\sqrt{\sum_{i=1}^{n} A_i^2}\sqrt{\sum_{i=1}^{n} B_i^2}},$$

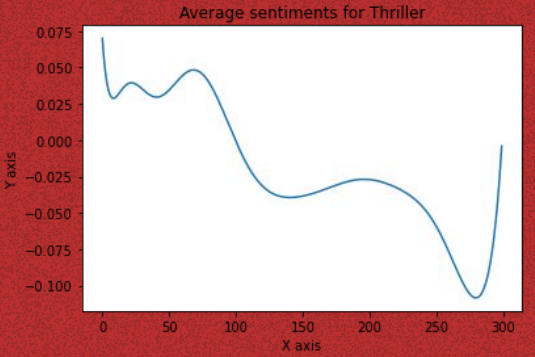
# RESULTS: RECOMMENDER

Ex: For a user who likes **Mission Impossible (Action/Adventure)**, **Oblivion (Action),** and **Indiana Jones and the Last Crusade (Adventure/Action)**, the top 10 recommended movies are:

1. Cliffhanger (Action/Adventure)
2. Someone to Watch Over Me (Thriller/Crime)
3. The Last of the Mohicans (War/Adventure)
4. Mute Witness (Crime/Horror)
5. Scott Pilgrim vs. The World (Action/Romance)
6. The Butterfly Effect (Sci-fi/Thriller)
7. Hollowman (Sci-fi/Horror)
8. 8mm (Thriller/Mystery)
9. Rear Window (Mystery)
10. Ghost Ship (Horror)

| Cluster | Genres Included |
|---|---|
| Cluster 1 | Short, Drama Comedy, Romance, Family, Music, Fantasy, Sport, Musical |
| Cluster 2 | Thriller, Horror, Action, Crime, Adventure, Sci-Fi, Mystery, Animation, Western |
| Cluster 3 | Documentary, History Biography, War, News |
| Cluster 4 | Reality-TV, Game Show, Talk Show |
| Cluster 5 | Adult |



# FUTURE WORK

- Incorporation of more features in script analysis to create higher-dimensional vectors for clustering
  - ex: profanity count, number of main characters, etc.
- Further analysis of script analysis graphs
  - ex: number of local maximums/minimums, first/second derivatives, overall change in tone from start to finish of movie
- Adding features beyond the script to create a more holistic recommender (more akin to existing systems)

# CONCLUSIONS

- Lots of information can be extracted from script analysis alone, without any visual/audio aspects of film
- Large scripts provide wealth of information, allowing for precise analysis that allows us to generate sentiment graphs that reflect our expectations for a given movie/genre
- Many genres share commonalities in script features, meaning a text-based recommender alone is capable of recommending similar movies that aren’t bound to a particular genre

# ROLES AND RESPONSIBILITIES

Jonny: Script sentiment analysis, Recommendation system

Ryan: Script Keyword Clustering Task, research on related movie clustering tasks, Future Work/Conclusions

Tanzila: finding dataset for clustering, features for sentiment analysis, literature review on movie sentiment analysis and movie recommenders, Recommender