MARCUS: An Event-Centric NLP Pipeline that generates Character Arcs from Narratives

Sriharsh Bhyravajjula[1], Ujwal Narayan[1] and Manish Shrivastava[1]

[1]*International Institute of Information Technology, Hyderabad*

Abstract

Character arcs are important theoretical devices employed in literary studies to understand character journeys, identify tropes across literary genres, and establish similarities between narratives. This work addresses the novel task of computationally generating event-centric, relation-based character arcs from narratives. Providing a quantitative representation for arcs brings tangibility to a theoretical concept and paves the way for subsequent applications. We present MARCUS (Modelling Arcs for Understanding Stories), an NLP pipeline that extracts events, participant characters, implied emotion, and sentiment to model inter-character relations. MARCUS tracks and aggregates these relations across the narrative to generate character arcs as graphical plots. We generate character arcs from two extended fantasy series, Harry Potter and Lord of the Rings. We evaluate our approach before outlining existing challenges, suggesting applications of our pipeline, and discussing future work.

Keywords

Computational Literary Studies, Character Arcs, Relations, Events, Narratives

1. Introduction

Characters in narratives attempt to influence their circumstances to resolve conflict, while circumstance itself shapes the characters with events that develop them [1]. This character journey is integral to narratives [2]; good stories are driven by the transformative journeys of compelling characters. This work addresses the challenge of quantifying these journeys using character arcs modeled around events and relations.

The field of computational literary studies aims to understand, represent and generate narratives. Existing research has focused on various perspectives, notably plot units [3, 4], social network extraction from narratives [5], and character-centric approaches [6, 7]. This work builds upon the latter to extract characters from narratives using co-reference resolution and clustering techniques. We find inspiration in research exploring events in narratives [8] and argue that the transformation of characters is a consequence of events that involve these characters. We also take advantage of recent advances in semantic role labeling [9] to understand the function of characters in events and build on advancements in emotion analysis [10, 11] to represent how events affect both agent and recipient characters quantitatively.
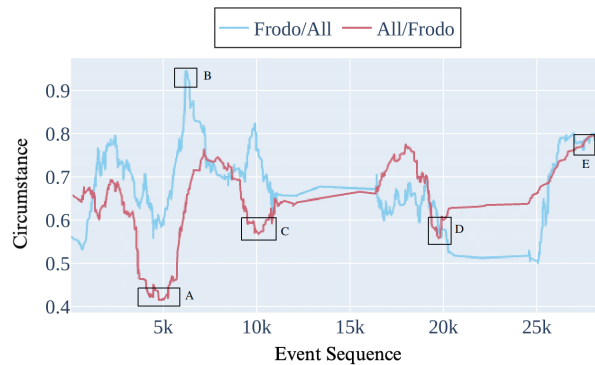
Figure 1: Relation-Based Character Arc for Frodo Baggins across The Lord of the Rings trilogy. The blue line represents Frodo the *actor*, and the red line represents Frodo the *experiencer*. **A**: Frodo is stabbed by the poisoned blade of the Nazgul at Weathertop (valley). **B**: Frodo reunites with loved ones at Rivendell (peak). **C**: Frodo is in grief after the wizard Gandalf falls to a Balrog (valley). **D**: Frodo is attacked by Shelob in her lair (valley). **E**: Frodo succeeds in his quest and returns home to The Shire (peak).

Our approach differs from recent work in the closely related field of extracting relationships between characters in terms of goals and linguistic features. Some existing approaches either lack the nuance of emotions [12, 13] or do not leverage semantic role labels for characters [14]. Other approaches fail to utilize fine-grained sentiments [15] or do not share our event-based perspective on relationships [16, 17]. While there has been progress in the field of emotion arcs, it has been centered around plot rather than character, capturing a shift in sentiment rather than a shift in circumstance [18, 19]. To the best of our knowledge, we believe that our task and approach are novel additions to the field.

2. Key Concepts

2.1. Events

We borrow from [8] and focus solely on *events* with asserted *realis* (depicted as actually taking place, with specific participants at a specific time) instead of those with other epistemic modalities (hypotheticals, future events). We interpret *events* as indicators of inter-character relationship states - for each event, we extract the latent predicate-argument structure to identify participants. For example, in the sentence "Harry kicked Ron", "kicked" is the *event* with Harry the *actor* and Ron the *experiencer*.

2.2. Circumstance

We take inspiration from [2] and use implied sentiment and emotion to quantify *circumstance*, the state of fortune associated with a directed pair of characters participating in an *event*. Narratives and plots span a plethora of settings - each character 'reacts to' as well as 'influences' their unique *circumstance*; it would be beyond the scope of this paper to establish a universal scale. Instead, we focus on the relative *shift of circumstance* across the narrative.

Table 1

Dataset Details

Data Source	Word Count	Event Count
Harry Potter	1,095,940	93,782
Lord of the Rings	478,329	28,670

Table 2

RoBERTa Sentiment Regression Model

Metric	Score
Mean Squared Error	0.01620
Mean Absolute Error	0.09693

2.3. Relation Arcs and Character Arcs

As the story progresses, the relationship between a pair of characters evolves. A *relation arc* is a plot of the shift in circumstance of a directed pair of characters across their participatory events in a narrative. While the *actor* of an event influences the circumstances that affect the *experiencer*, the latter's subsequent actions are captured in other *relation arcs*, allowing the effect of events to trickle through multiple connected arcs forming the plot of a narrative. We posit that a character's journey at any point in the narrative can be represented by contextually assessing the amalgamated effect of their interactions with both themselves and other characters on their circumstance; a *character arc* is thus defined as a pair of quantitative aggregations of all the corresponding *relation arcs* of a character as both *actor* and *experiencer* respectively.

Fig 1 shows the character arc for the protagonist of the Lord of the Rings, Frodo Baggins, capturing the shift of circumstances he experiences as both an *actor* and *experiencer*. The events Frodo participates in change his circumstances - a drop to the valleys (minimas) represents a deterioration of circumstances, whereas a rise to the peaks (maximas) denotes an improvement. It is evident from **C** and the arc slice between **D** and **E** that actor and experiencer arcs of a character are not always aligned, allowing for the intuitive explanation of instances when characters oppose the circumstances they find themselves in, for better or for worse.

3. MARCUS

We present MARCUS (Modelling Arcs for Understanding Stories), an event-centric NLP pipeline[1] that generates character arcs from narratives. In this section, we will elaborate upon the various components of the pipeline.

3.1. Dataset

We failed to retrieve meaningful character arcs in our preliminary experiments with short stories containing less than 500 events. Consequently, we choose two larger datasets for character arc analysis instead, namely The Harry Potter septology and The Lord of the Rings trilogy. For each source, we ensure that we remove page numbers, unnecessary annotations, unreadable characters, weblinks, etc. The details of each dataset are detailed in Table 1.
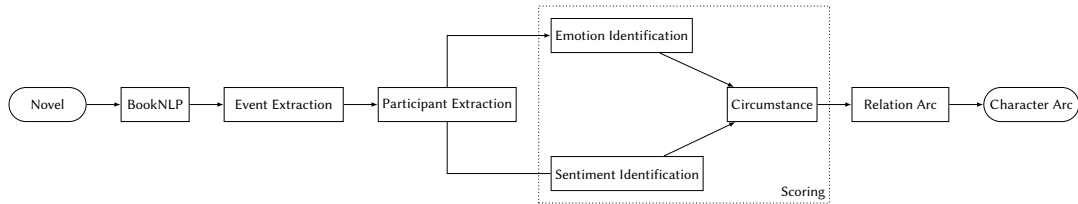
---

[1]https://github.com/darthbhyrava/MARCUS

Figure 2: MARCUS (Modeling Arcs for Understanding Stories), an NLP pipeline that plots a character's arc as their quantitative interaction with circumstance as both actor and experiencer, represented by the proxy amalgamation of their event-centric relations across the narrative.

3.2. BookNLP

BookNLP [7] is an NLP pipeline[2] that scales to books and long documents (in English) and performs tasks like part-of-speech tagging (Stanford), dependency parsing (MaltParser), named entity recognition (Stanford), character name clustering (e.g., "Tom", "Tom Sawyer", "Mr. Sawyer" : "Tom Sawyer") and pronominal coreference resolution. MARCUS uses BookNLP to retrieve character occurrences and linguistic features needed for event and participant extraction.

3.3. Event Extraction

We feed the features processed from BookNLP into a tagger consisting of a BiLSTM model with BERT embeddings to extract events and corresponding entities from the narrative. The tagger is trained[3] using the LitEvents dataset [8], which provides an annotated list of events from over 100 public domain novels from Project Gutenberg. MARCUS uses tagged events and entities for extracting participant characters, sentiment, and implied emotion.

3.4. Participant Extraction

We process each event with a BERT-based Semantic Role Labeller provided by AllenNLP[4] to extract the latent predicate-argument structure of the event and identify roles of participant characters. If there are multiple events in the same sentence with the same actors and experiencers, we consider only the first event to avoid redundancy.

3.5. Circumstance

Every actor/experiencer pair corresponding to an event in the narrative is assigned a quantitative measure of circumstance. We argue that proxy indicators such as sentiment and implied emotion implicitly capture an event's circumstances and are specifically well suited to our focus on *shift of circumstance*. Absolute measures of circumstance can therefore be interpreted as a characteristic of the genre or trope of narrative, while their shifts are a characteristic of the character's journey. Since circumstances evolve over time (or event sequences), MARCUS considers the effect of previous circumstances between characters when defining *relation arcs*.

---

[2]https://github.com/dbamman/book-nlp
[3]https://github.com/dbamman/litbank
[4]https://docs.allennlp.org/models/main/models/structured_prediction/models/srl_bert/

3.5.1. Sentiment Identification

We pose sentiment extraction as a regression task to capture the subtleties of relationships. We fine-tune a RoBERTa model on the Stanford Sentiment Treebank (SST) [20] dataset to obtain a fine-grained sentiment score in the range of 0 to 1. The SST dataset provides $12k$ sentences and phrases with their associated sentiment scores lying between 0 to 26, which we normalize before training for ten epochs in a 60:20:20 split. The metrics for this model are reported in Table 2. MARCUS uses the model to assign sentiment scores to events extracted earlier in Section 3.3.

3.5.2. Emotion Identification

Sentiment alone may not give us enough information about circumstance - we argue that in such cases, multi-faceted emotional states help capture shift in circumstance by leveraging the nuances of relationships. To identify emotions, we use a BERT model[5] trained on the GoEmotions [10] dataset in a multi-label setting, as interactions can have more than one emotional undertone. The GoEmotions dataset consists of 58k Reddit comments manually annotated for 27 emotion categories: admiration, amusement, sorrow, fear, etc. MARCUS uses the confidence of the model's predicted labels as well as a manually assigned value for each label to contribute to the measure of circumstance. These labels, in the range of -2 to 2, are assigned based on intensity of emotion; higher intensity corresponds to higher absolute value.

3.6. Relation and Character Arcs

MARCUS generates relation arcs by plotting the measure of circumstance, t, for every event, e, belonging to an actor/experiencer pair of characters across the narrative.

$$t_e = \alpha * s_e + \sum_{i=1}^{L} \beta_i * c_{i_e} \tag{1}$$

where t_e is the measure of circumstance for event e, $\alpha \in (0, 1)$ is the sentiment co-efficient that controls how much influence the fine grained sentiment score should have over relation arcs, $s_e \in (0, 1)$ is the sentiment of that event, L is the total number of emotion labels for that event, $\beta_i \in [-2, 2]$ is the fixed score for emotion label, and $c_{i_e} \in (0, 1)$ is the corresponding confidence score for each emotion label in the event. We run multiple experiments to choose optimal values of α and β: their final values are listed in the code.

We apply a window function $R(.)$ over $\mathbf{t_e}$, the set of all measures of circumstance corresponding to the event set $\mathbf{e}$, to calculate the relation arc, $\mathbf{r}$, given by

$$\mathbf{r} = R(\mathbf{t_e}, n, p) \tag{2}$$

where $R(.)$ is the window function that helps smoothen the arcs while retaining previous state information, n is the window size, and p is an optional parameter for specifying order for polynomial fitting.

As shown in Fig 3a, the relation arcs are too noisy without smoothing or retention of previous circumstance information. We experiment with three standard window functions: Rolling
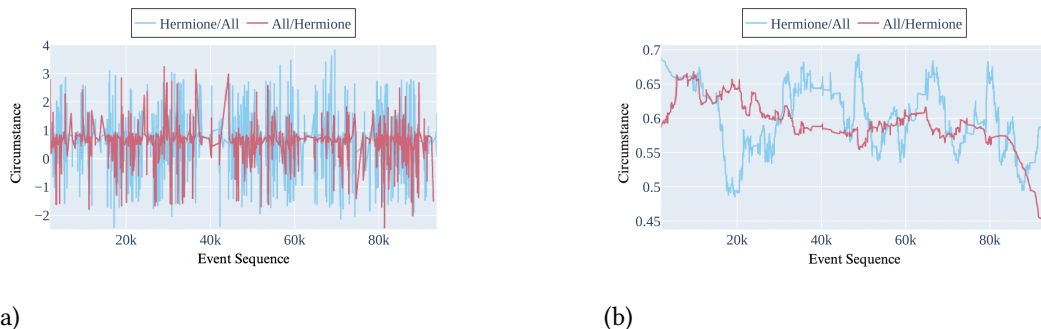
---

[5]https://github.com/monologg/GoEmotions-pytorch

(a)   (b)

Figure 3: (a) Character Arc for Hermione, No Rolling Function Applied; (b) Character Arc for Hermione, with a savgol filter of window size 1/10th of her event sequence length, fitted with a third degree polynomial.

Mean, Triangular Rolling Mean and Savitzky-Golay Filter [21]. We find that the Savitzky-Golay Filter represents the narrative most accurately as seen in Fig 3b, and use the same for all arcs represented in the paper. We generate the character arc, **c**, by adding up the corresponding relation arcs **r** of that character over all events **e** in the role of actor and experiencer respectively.

4. Evaluation

4.1. Survey

We ask a set of 16 human volunteers (avid fiction readers aged 20-31) to peruse both the Harry Potter and Lord of the Rings series, following which they evaluate our system by answering surveys on two tasks: **idchar**, where the volunteers are given a list of relation arcs and character pairs and asked to match the arcs to their corresponding pairs, and **idplot**, where the volunteers are given character arcs, pertinent plot events and asked to identify the points in the arc that they think represent the corresponding events.

For both these tasks mentioned above, we calculate accuracy. Task **idchar** achieved an accuracy of 71.5% and task **idplot** had an accuracy of 72.9%. We also evaluate with the Fleiss Kappa metric where category 1 indicates a complete match, and 0 indicates otherwise. For both the tasks, we have a score of 0.675 and 0.528 indicating strong and moderate inter-annotator agreement, respectively. Thus, most of the volunteers consistently identified both the character pairs and the relevant points in the graph given the plot sequence.

4.2. Gold Labels

We have a volunteer extremely familiar with the story annotate the first 300 events of Lord of the Rings trilogy involving Frodo Baggins as a participant character. The annotator marks each event with three labels denoting a *positive, neutral* or *negative* shift in circumstance. Tables 3 and 4 illustrate the positive and negative shifts as tagged by our system and the corresponding gold labels for the same events provided by our annotator. Our system tends to assign positive labels to neutral events and has higher accuracy for negative shifts of circumstance.

Table 3
Positive Shifts

Gold Label	Percentage
Positive	0.36
Neutral	0.30
Negative	0.33

Table 4
Negative Shifts

Gold Label	Percentage
Positive	0.12
Neutral	0.15
Negative	0.73

5. Challenges and Future Work

We identify five notable challenges in our approach that can be addressed in future work. Firstly, since MARCUS is a sequential pipeline, it is challenging to determine the effect of errors cascading through the system quantitatively. Secondly, our rolling window makes the arcs dependent on the availability of data; event paucity in short stories or characters with low interactions hinders accurate arc generation. Thirdly, we observe in our arcs that our fine-grained events do not represent an abstract view of the discourse - a more contextual representation of events is needed. Fourth, in our understanding of a character's circumstance, localized interactions with other event-specific characters heavily influence shifts; we need an effective means of capturing the latent relative importance of character-specific interactions. And lastly, our approach does not aim to handle non-linear narratives where events are not sequentially presented.

6. Applications

Providing tangibility to the theoretical concept of character arcs, MARCUS can be employed in a variety of applications. Character arcs can be used in a more nuanced approach for detecting similarity between narratives by focusing on character journeys, leading to a possible improvement in book recommendations and movie recommendations based on stories and scripts. Character arcs can also help with digital enrichment in e-readers, adding to the rich metadata provided by devices like Kindle. Character arcs can also function as guidance for natural language generation tasks in the field of fiction. And lastly, they can help narrative studies by identifying character tropes and for identification of personality traits.

7. Conclusion

We propose MARCUS (Modeling Arcs for Understanding Stories), an NLP pipeline that addresses the novel task of generating character arcs from narratives. We explain key concepts like *events* and *circumstance* and delve into the details of our event-centric approach which leverages proxy markers like sentiment and emotion. We then evaluate our pipeline, discuss challenges, elucidate future work and outline potential applications.

References

[1] K. M. Weiland, Creating Character Arcs, The Masterful Author's Guide to Uniting Story Structure, Plot, and Character Development, PenForASword Publishing, 2016.

[2] K. Vonnegut, Kurt Vonnegut on the Shapes of Stories, YouTube, 1995. URL: https://youtu.be/oP3c1h8v2ZQ.

[3] M. Elsner, Character-based kernels for novelistic plot structure, EACL, 2012.

[4] W. Lehnert, Plot units and narrative summarization, Cogn. Sci. 5 (1981) 293–331.

[5] A. Agarwal, A. Kotalwar, J. Zheng, O. Rambow, SINNET: Social interaction network extractor from text, IJCNLP (System Demonstrations), 2013.

[6] L. Flekova, I. Gurevych, Personality profiling of fictional characters using sense-level links between lexical resources, EMNLP, 2015.

[7] D. Bamman, T. Underwood, N. A. Smith, A Bayesian mixed effects model of literary character, ACL, 2014.

[8] M. Sims, J. H. Park, D. Bamman, Literary event detection, ACL, 2019.

[9] P. Shi, J. J. Lin, Simple bert models for relation extraction and semantic role labeling (2019).

[10] D. Demszky, D. Movshovitz-Attias, J. Ko, A. S. Cowen, G. Nemade, S. Ravi, Goemotions: A dataset of fine-grained emotions, ACL (2020).

[11] S. Zad, M. Finlayson, Systematic evaluation of a framework for unsupervised emotion recognition for narrative text, in: Proceedings of the First Joint Workshop on Narrative Understanding, Storylines, and Events, ACL, 2020, pp. 26–37.

[12] S. Srivastava, S. Chaturvedi, T. M. Mitchell, Inferring interpersonal relations in narrative summaries, CoRR abs/1512.00112 (2015).

[13] S. Chaturvedi, S. Srivastava, H. D. III, C. Dyer, Modeling dynamic relationships between characters in literary novels, CoRR abs/1511.09376 (2015).

[14] M. Iyyer, A. Guha, S. Chaturvedi, J. Boyd-Graber, H. Daumé III, Feuding families and former Friends: Unsupervised learning for dynamic fictional relationships, NAACL-HLT, ACL, 2016.

[15] F. Rashid, E. Blanco, Characterizing interactions and relationships between people, EMNLP, 2018.

[16] E. Kim, R. Klinger, Frowning frodo, wincing leia, and a seriously great friendship: Learning to classify emotional relationships of fictional characters, CoRR abs/1903.12453 (2019).

[17] S. Chaturvedi, M. Iyyer, H. D. III, Unsupervised learning of evolving relationships between literary characters, AAAI, 2017.

[18] S. Somasundaran, X. Chen, M. Flor, Emotion arcs of student narratives, in: Proceedings of the First Joint Workshop on Narrative Understanding, Storylines, and Events, ACL, 2020.

[19] A. J. Reagan, L. Mitchell, D. Kiley, C. M. Danforth, P. S. Dodds, The emotional arcs of stories are dominated by six basic shapes, CoRR abs/1606.07772 (2016).

[20] R. Socher, A. Perelygin, J. Wu, J. Chuang, C. D. Manning, A. Ng, C. Potts, Recursive deep models for semantic compositionality over a sentiment treebank, EMNLP, 2013.

[21] A. Savitzky, M. J. Golay, Smoothing and differentiation of data by simplified least squares procedures., Analytical chemistry 36 (1964) 1627–1639.