MBTI Personality Prediction for Fictional Characters Using Movie Scripts

Yisi Sang1∗ Xiangyang Mou2∗ Mo Yu3∗ Dakuo Wang4 Jing Li5 Jeffrey Stanton1
1Syracuse University 2Rensselaer Polytechnic Institute 3Pattern Recognition Center, WeChat AI
4IBM Research, Northeastern University 5New Jersey Institute of Technology
yisang@syr.edu moux4@rpi.edu moyumyu@tencent.com

Abstract
An NLP model that understands stories should be able to understand the characters in them. To support the development of neural models for this purpose, we construct a benchmark, Story2Personality. The task is to predict a movie character’s MBTI or Big 5 personality types based on the narratives of the character. Experiments show that our task is challenging for the existing text classification models, as none is able to largely outperform random guesses. We further proposed a multi-view model for personality prediction using both verbal and non-verbal descriptions, which gives improvement compared to using only verbal descriptions. The uniqueness and challenges in our dataset call for the development of narrative comprehension techniques from the perspective of understanding characters.1

1 Introduction
Character comprehension is commonly regarded as the cornerstone to comprehending stories in psychology and education (Bower and Morrow, 1990; Paris and Paris, 2003; Zhao et al., 2022). The NLP community has done some work on character comprehension in reading comprehension tasks, but most of the existing studies focus on short or expository texts (e.g., story summaries) (Urbanek et al., 2019; Brahman et al., 2021). Moreover, most of them are limited in factoid understanding of characters, such as coreference resolution (Chen and Choi, 2016) and character relationships (Iyyer et al., 2016), and few studies have explored deeper comprehension of characters’ persona (Flekova and Gurevych, 2015; Sang et al., 2022a), on which humans can generally do well.

We propose Story2Personality, a new narrative understanding benchmark to encourage the study of character understanding. The goal of Story2Personality is to predict personality according to the character’s narrative texts in the script.

Personality prediction from narratives has many challenges. First, stories often use a variety of narrative clues (e.g., scenery changes), sequence (e.g., flashback) and rhetorical techniques (e.g., metaphor) (Xu et al., 2022b). Second, the inputs of the task are long (>10K words on average), challenging the applications of Transformer-based models (Vaswani et al., 2017). Third, both the scene descriptions and dialogues are informative for the prediction, requiring models to jointly consider multiple views of inputs.

This study makes the following contributions:
• We establish a large-scale dataset for personality prediction of narrative characters that can support the development of neural models. Our dataset consists of 3,543 characters from 507 movies with MBTI labels of four dimensions. In comparison, the only existing related dataset (Flekova
and Gurevych, 2015) contains only 298 characters and focuses on a single dimension. Our dataset is proved challenging — on this binary classification task, none of the baselines achieve higher than 60% macro-F1.

- We develop a movie script parser to automatically process a script to a structured form with the verbal character dialogues and the non-verbal scene descriptions illustrating backgrounds. Human study shows that our parser is more accurate compared to previous rule-based tools.
- We propose an extension to BERT classifier (Devlin et al., 2018) to handle the long and multi-view (verbal and non-verbal) inputs. Our model improves 2-3% over the baselines. This shows the potential of exploiting both verbal and non-verbal narratives of characters, which is consistent with psychological theory (McCroskey and Richmond, 1996; Richmond et al., 2008); and suggests directions of future model design.

2 Related Work

Character-Centric Narrative Understanding

There have been existing studies on character-centric narrative understanding. While many of them (Massey et al., 2015; Srivastava et al., 2016; Brahman et al., 2021) work on summaries of stories or summaries of characters. Their scopes thus have a different assessment purpose from ours, and have the challenge on understanding long narrative inputs greatly reduced.

For works that use long narratives, most of them study the inter-character relationship (Elson et al., 2010; Elsner, 2012; Elangovan and Eisenstein, 2015; Iyyer et al., 2016; Chaturvedi et al., 2016, 2017; Kim and Klinger, 2019). Inter-character relationship is also related to social network theories. Various of relationships have been considered in these studies, while most of them rely on unsupervised learning and do not provide labeled data for a direct automatic evaluation. TVSHOWGUESS explored multiple perspectives of persona using long narratives but the task format is different from us (Sang et al., 2022b).

Finally, there is work on fundamental NLP annotating techniques over books and screenplays, such as named entity recognition (Bamman et al., 2019), coreference resolution (Chen and Choi, 2016), event-centric extraction (Xu et al., 2022a), and entity-centric natural language modeling (Clark et al., 2018) which is different from narrative understanding. Their techniques can be helpful to our task but the scope of their research is different from character-centric comprehension.

Latent Persona Induction

Besides (Flekova and Gurevych, 2015) that is similar to our work in terms of the focus on personality classification, there is another line of related work on latent persona induction (Bamman et al., 2013). The work learns a topic model over character behaviors from books, and each latent topic corresponds to an induced persona. The induced persona vectors can be then applied to potential applications as a type of character representation.

From the perspective of practicality, our work and (Bamman et al., 2013) have their own strengths. From our motivation of story comprehension assessment, the difference is whether we provide a direct evaluation of the character understanding or evaluate it in down-streaming tasks – similar to the aforementioned relationship detection work, it is also difficult to provide an automatic and objective evaluation for the task of (Bamman et al., 2013). The advantage of our task is that it supports direct automatic evaluation by itself, without the need for further downstream tasks; and it can be also used to evaluate the methods for the task of (Bamman et al., 2013). Moreover, compared to a direct evaluation, the performance on a down-streaming task can be affected by other factors other than persona so a good performance on downstream tasks may not come directly caused by a good persona representation. The cons of our task is that it is limited to the personality types that have human annotations.

3 Background of MBTI

Personality is a “stable and measurable” individual characteristic (Vinciarelli and Mohammad, 2014) which can “distinguish internal properties of the person from overt behaviors” (Matthews et al., 2003). Understanding the personalities of the characters is essential for grasping the story’s greater message. The Myers–Briggs Type Indicator (MBTI) (Myers, 1962) and the Big-5 Personality are two of the most popular personality scales. We used MBTI as the annotation criteria since despite some validity controversy in self-report measurement, research shows that a person’s friend can accurately judge his/her MBTI personality (Cohen et al., 1981). In our narrative comprehension scenario, a fictional character’s MBTI personality is judged by other human raters in an online com-
munity, which is quite similar to the third-person evaluation scenario, and should yield a reasonable validity. We also conducted our study on Big-5 and reported the results in Appendix 6.

MBTI assess the psychological preferences in how people perceive the world and make decisions in four dimensions: E/I: extravert (E) is seen as being generally active and objective while the introvert (I) is seen as generally passive and subjective (Sipps and Alexander, 1987). S/N: sensing (S) is seen as attending to sensory stimuli; intuition (N) describes a more detached, insightful analysis of events and stimuli (Boyle, 1995). T/F: thinking (T) involves logical reasoning and decision making; feeling (F) involves a more subjective and interpersonal approach (Thomas, 1983). J/P: judging (J) attitude is associated with prompt decision making; perception (P) involves greater patience and waiting for more information before making a decision. An individual’s MBTI type has a label based on her dominant preference for each dimension. In Figure 1, Morpheus is an extraversion person, understanding the world with intuition, dealing with things with feeling, and organize the world around him by judging. Together gives an ENFJ type.

4 Story2Personality Dataset

We constructed our dataset in three stages: extracting movie scripts from the Internet Movie Script Database (IMSDB ²), parsing the collected movie scripts into dialogue and scene sections, matching characters’ personality types from The Personality Database (PDB³) with their dialogues and scenes.

4.1 Movie Scripts Collection

We collected HTML files from IMSDB combined with movie scripts in NarrativeQA (Kočiský et al., 2018). After removing corrupted or empty files, we got 1,464 usable scripts.

4.2 Our Statistical Movie Script Parser

As shown in Figure 1, a movie script usually has four basic format elements (Riley, 2009): Scene Headings, one line description of each scene’s type, location, and time (i.e., INT. ROOM 1313); Scene description, the description of the actions of the characters (i.e., text in blue); Dialogues, names of characters and actual words they speak (i.e., text in red); Transitions, instructions for linking scenes together (i.e., FADE IN ON).

In order to extract dialogues and scene descriptions in a structured form, we first split the scripts to sections, i.e., text chunks between two adjacent bolded chunks which are scene headings or character names and stored the bolded texts as section titles. Then we designed a statistical method to classify the section types:

Rule-Based Pre-Processing We start with a rule to classify the sections into dialogues and scenes. As Figure 1 shows, a common format of movie scripts is to align the shot headings, transitions and scene descriptions vertically, and uses a larger indentat for dialogues. So, the indent size can be used to identify dialogues. Since the indentat size may vary across different scripts. Our rule assumes the sections as dialogues if they have larger indent compared to FADE IN in the same script and the others as scenes.

Silver Parses Construction The rule-based pre-processing introduces many noises. We then designed a statistical method to automatically determine the threshold indent of dialogues. First, we compute the averaged ratio of dialogues in a script and its standard variation σ. Second, we keep adding sections with the largest indent sizes to the set of dialogues, until the ratio of added sections becomes larger than μ+σ. Finally, we keep the left sections as scenes. If none of the indentation size can reach the ratio of dialogues in the range of μ±σ, the movie script was seen as a failure case. We designated the successfully processed scripts with the dialogues/scene labels as the “silver” set which consists of 29% of the scripts.

Section Classifier For the failure scripts from the previous step and the scripts without FADE IN markers, we trained a BERT-based section classifier using 137,042 labeled sections from the silver set to label them. The classifier achieved 99.31% accuracy on a held out validation set. The outputs are our final parses.

4.3 Personality Collection and Mapping

We collect human rated MBTI types from PDB. Movie scripts are the blueprint for the actor’s performance. An actor’s body language, dialogue, and contexts are all described in the scripts (Jhala, 2008). Human rater’s perception of a character’s personality from the movies would be consistent

²https://imsdb.com/
³https://www.personality-database.com/
with the script’s description. In total, we collected MBTI types of 28,653 characters. Each character has an id, name, vote count, and voters’ agreement on each MBTI dimension. For example, the MBTI profile in Figure 1 has 300 voters, with different agreement rate along each dimension. To ensure the quality of personality voting, we removed character profiles with <3 voters and <60% agreement rate so some characters do not have all the 4 dimensions. We include more details in Table 5 in the appendix. PDB’s When the user starts rating, the rating interface hides the previous rater’s choices. Thus, the rater would not have prior bias. We then matched the characters’ personality profiles to the scripts, if the name can be softly matched to the dialogue title or the recognized named entities in the scenes (details and example of the final processed data in Appendix A). Table 1 shows the core statistics of our dataset. The numbers of data points can also be found in Table 6.

5 Dataset Analysis

We conduct human study to verify the advantage of our script parser; then provide the human performance on our dataset.

Script Parsing results We compared our parsing results with the results of the state-of-the-art opensourced script parser (Ramakrishna et al., 2017), which employs many human written rules, with a human study. We randomly selected five scene descriptions and five dialogue sections in 10 common movies, giving 100 snippets for evaluation (40 from the silver set). Then we manually compared the parsing results with the original movie scripts. Table 2 shows the results. Our parser outperforms Ramakrishna et al. (2017) with a large margin. Most mistakes of (Ramakrishna et al., 2017) is to recog-

6 Experiments

Baselines We build two baseline models. • SVM, the LinearSVC from sklearn.svm. We extracted top 20K word unigram, bigram, and trigram features according to term frequency after removing stop words. We set $C=0.1$. • BERT, fine-tuning the out-of-box BERT, with a linear head on the ‘[CLS]’ token’s final layer embedding for classification.

Our Method We propose the multi-view multi-row BERT (MV-MR BERT) classifier (Fig. 2)
which is an extension of BERT to deal with the long inputs and handle the verbal and non-verbal information differently. First, to handle the long input per character, we borrow the idea from fusion-in-decoder (Izacard and Grave, 2020). Since the complexity of Transformers is $O(RL^2)$ (with $R$ the number of rows and $L$ the length per row), when $L$ is very large, we can split it into multiple segments to reduce the quadratic term. Next, we rely on the attention over all the segments to fuse the information. Specifically, we split the input content $D$ of a character into multiple segments $D = \{S_i\}_{i=1}^R$, and encode all the segments in a minibatch as $H = \text{BERT}(S_i) \in \mathbb{R}^{R \times L \times d}$, where $d$ is the hidden state size. Then a linear head is applied to get the attention score across tokens in all the rows as $\alpha = \text{softmax}(HW + b) \in [0, 1]^{R \times L}$. The final summarized representation of the input $D$ is thus the weighted summation $h_D = \sum_{i=1}^R \sum_{j=1}^L \alpha_{ij} H_{ij}$. Second, to handle both the dialogue and behavioral description a character, our multi-view model receives an input pair $(D^{\text{dialog}}, D^{\text{scene}})$, then uses a shared BERT and separated linear heads to compute the summarized states $h_D^{\text{dialog}}$ and $h_D^{\text{scene}}$. The two vectors are fed into a fully-connected layer for prediction. For the scene descriptions, we prepend a special token “[ent]” to the target character’s name to denote its position. The attention $\alpha^{\text{scene}}$ is only computed on these special tokens.

Results and Analysis Following Flekova and Gurevych (2015), we use macro-averaged F1 as evaluation metric. Table 3 shows the main results on the four MBTI dimensions. Peak performance was achieved by our MV-MR BERT. The result suggests using both dialog and action scene descriptions consistently improved model performance.

The results are generally low compared to human performance, showing the task is challenging to existing models. We analyzed the learning curve of BERT model by adding the training data from 1K to 2.5K characters (Table 7 in Appendix D). The model performance did not change a lot in the development dataset. Figure 3 in Appendix D gives further evidence for the challenge of our task, which shows the dev and test results are not highly-correlated, meaning that by achieving near perfect accuracy on the training data, the models largely overfit the noises instead of capturing real clues.

Model Performance on the Big Five Personality Test We collected a variation of Big 5, the Global 5, from the PDB. The Global 5 adaptation of the Big Five (Digman, 1997) consists of Extraversion, Emotional Stability, Orderliness, Accommodation, and Intellect. The SLOAN nation system is the scoring format for Global 5 test. SLOAN nation keys are: Social/Reserved, Limbic/Calm, Organized/Unstructured, Accommodating/Egocentric, Non-curious/Inquisitive. The number of people who evaluated the movie characters with big5 was very small, and after eliminating the characters without dialogues, we only got 1,346 characters’ data. Such a small amount of data deeply affects the training of neural-based model. Table 4 shows the model performance.

Limitation Movie scripts are the blueprint for the actor’s performance. An actor’s dialogue, body language, and the contexts are well described in the scripts. There is sufficient information in the scripts for readers to understand the characters. However, the actors’ portrayal have the potential to add additional cues to influence the audience’s perception of the fictional characters’ personalities. In future work we will try to use multi-modality data as input.

Conclusion We develop a movie script parser and proposed a new narrative understanding benchmark, Story2Personality, which enables neural model training for understanding characters. We evaluate several classifiers on our task – while our multi-view multi-view BERT model achieves a substantial improvement over the SVM and BERT baselines, there is a huge gap compared to human performance. This indicates our dataset a valuable and challenging task for future research. In the future we will expand our dataset.
References

Apoorv Agarwal, Sriramkumar Balasubramanian, Jiehan Zheng, and Sarthak Dash. 2014. Parsing screenplays for extracting social networks from movies. In Proceedings of the 3rd Workshop on Computational Linguistics for Literature (CLFL), pages 50–58.

David Bamman, Brendan O’Connor, and Noah A Smith. 2013. Learning latent personas of film characters. In Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), pages 352–361.

David Bamman, Sejal Popat, and Sheng Shen. 2019. An annotated dataset of literary entities. In Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers), pages 2138–2144.

Gordon H Bower and Daniel G Morrow. 1990. Mental models in narrative comprehension. Science, 247(4938):44–48.

Gregory J Boyle. 1995. Myer-briggs type indicator (mbti): some psychometric limitations. Australian Psychologist, 30(1):71–74.

Faeze Brahman, Meng Huang, Oyvind Tafjord, Chao Zhao, Minmaya Sachan, and Snigdha Chaturvedi. 2021. “let your characters tell their story”: A dataset for character-centric narrative understanding. arXiv preprint arXiv:2109.05438.

Snigdha Chaturvedi, Mohit Iyyer, and Hal Daume III. 2017. Unsupervised learning of evolving relationships between literary characters. In Proceedings of the AAAI Conference on Artificial Intelligence, volume 31.

Snigdha Chaturvedi, Shashank Srivastava, Hal Daume III, and Chris Dyer. 2016. Modeling evolving relationships between characters in literary novels. In Proceedings of the AAAI Conference on Artificial Intelligence, volume 30.

Yu-Hsin Chen and Jinho D Choi. 2016. Character identification on multiparty conversation: Identifying mentions of characters in tv shows. In Proceedings of the 17th Annual Meeting of the Special Interest Group on Discourse and Dialogue, pages 90–100.

Elizabeth Clark, Yangfeng Ji, and Noah A Smith. 2018. Neural text generation in stories using entity representations as context. In Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers), pages 2250–2260.

David Cohen, Marilye Cohen, and Herbert Cross. 1981. A construct validity study of the myers-briggs type indicator. Educational and Psychological Measurement, 41(3):883–891.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. Bert: Pre-training of deep bidirectional transformers for language understanding. arXiv preprint arXiv:1810.04805.

John M Digman. 1997. Higher-order factors of the big five. Journal of personality and social psychology, 73(6):1246.

Vinodh Krishnan Elangovan and Jacob Eisenstein. 2015. “you’re mr. lebowski, i’m the dude”: Inducing address term formality in signed social networks. In HLT-NAACL.

Micha Elsner. 2012. Character-based kernels for novelistic plot structure. In Proceedings of the 13th Conference of the European Chapter of the Association for Computational Linguistics, pages 634–644.

David K Elson, Kathleen McKeown, and Nicholas J Dames. 2010. Extracting social networks from literary fiction.

Lucie Flekova and Iryna Gurevych. 2015. Personality profiling of fictional characters using sense-level links between lexical resources. In Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing, pages 1805–1816.

Mohit Iyyer, Anupam Guha, Snigdha Chaturvedi, Jordan Boyd-Graber, and Hal Daumé III. 2016. Feuding families and former friends: Unsupervised learning for dynamic fictional relationships. In Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, pages 1534–1544.

Gautier Izacard and Edouard Grave. 2020. Leveraging passage retrieval with generative models for open domain question answering. arXiv preprint arXiv:2007.01282.

Arvind Jhala. 2008. Exploiting structure and conventions of movie scripts for information retrieval and text mining. In Joint International Conference on Interactive Digital Storytelling, pages 210–213. Springer.

Carl Jung. 2016. Psychological types. Routledge.

Evgeny Kim and Roman Klinger. 2019. Frowning frodo, wincing leia, and a seriously great friendship: Learning to classify emotional relationships of fictional characters. In Proceedings of NAACL-HLT, pages 647–653.

Tomáš Kočiský, Jonathan Schwartz, Phil Blunsom, Chris Dyer, Karl Moritz Hermann, Gábor Melis, and Edward Grefenstette. 2018. The narrativeqa reading comprehension challenge. Transactions of the Association for Computational Linguistics, 6:317–328.

Philip Massey, Patrick Xia, David Bamman, and Noah A Smith. 2015. Annotating character relationships in literary texts. arXiv preprint arXiv:1512.00728.
A Details of Dataset Construction

Soft Name Matching Algorithm We created two movie-character dictionaries to associate the characters with the movies using the characters’ full names and their subcategories (i.e., movie names) in personality profile data, as well as section titles (i.e., character names or scene headings) and movie names in the movie script data. Then, we tokenized and lowercased the character names. We matched both the exact same full names and the intersections of tokens such as the first or last name of the full name when the movie names are matched. To identify a character’s scene descriptions, we extracted named entities from scene descriptions and then matched the characters and scenes based on their names using the same method. After matching the character name with the movie name, we store the MBTI personality, vote count, dialogue and scene descriptions into a dictionary for each character.

Example Data Item for One Character Figure 4 shows the example of information for one character Gary from the movie “Joker” in our Story2Personality, stored in json format. The data item contains the ID (‘id’), character name (‘mbti_profile’) and movie name (‘subcategory’) in the PDB website; together with the human voted MBTI types and the number of votes. Finally we save the dialogues of the character and the scenes he appears in two separated entries. For scenes, we save both the scene texts and the soft matched name mentions in the texts for the target character. The name mention is used to prepend the special tokens in our MV-MR BERT model.

B Additional Information of the Dataset

Table 5 lists the distribution of all the 16 MBTI types in our dataset, together with a representative movie character for each type.

C Statistics of Human Agreement

Table 6 lists the human agreement score on each MBTI dimension, on which we compute the human accuracy and approximate the human macro-F1 scores.

D Additional Model Performance

Figure 3 gives further evidence for the challenge of our task, which plots the dev versus test scores during our model selection. It shows the dev and test results are not highly-correlated, meaning that by achieving near perfect accuracy on the training
data, the models largely overfit the noises instead of capturing real clues. Both length and multiview have an improvement on model performance, but length has a slightly smaller impact, as shown in Table 8, when increasing the number of input tokens, the performance is not greatly affected.

|       | 0.5K  | 1K   | 2K   | 4K   |
|-------|-------|------|------|------|
| J/P   | 59.79 | 58.18| 60.35| 58.77|
| T/F   | 59.91 | 64.31| 63.32| 65.42|
| N/S   | 56.15 | 53.86| 55.65| 57.18|
| I/E   | 61.64 | 61.05| 62.87| 62.69|

Table 8: Ablation experiment on input length.

E  Model Checklist

We implement our baselines based on HuggingFace Transformers.4 We use the pre-trained bert-base-uncased models. We train all the models with the Adam optimizer.

We train our model on a single V100 GPU. It takes around 2 hour and 10 minutes to train a multi-row BERT model. For all the models, we train in total 20 epochs.

Hyperparameters  We set the number of rows in MV-MR BERT to 20, to maximize the usage of GPU memory. We set the learning rate to 2e-5. We report the average performance of five runs.

4https://github.com/huggingface/transformers

Table 7: Learning curve with varying amount of training data (on N/S).
Figure 4: An example excerpt from Story2Personality. Each character has ID (‘id’), character name (‘mbti_profile’), movie name (‘subcategory’), dialogue, and scene descriptions.