Inducing Semantic Micro-Clusters from Deep Multi-View Representations of Novels

Lea Frermann
University of Edinburgh*
l.frermann@ed.ac.uk

György Szarvas
Amazon Development Center Germany GmbH
szarvasg@amazon.de

Abstract
Automatically understanding the plot of novels is important both for informing literary scholarship and applications such as summarization or recommendation. Various models have addressed this task, but their evaluation has remained largely intrinsic and qualitative. Here, we propose a principled and scalable framework leveraging expert-provided semantic tags (e.g., mystery, pirates) to evaluate plot representations in an extrinsic fashion, assessing their ability to produce locally coherent groupings of novels (micro-clusters) in model space. We present a deep recurrent autoencoder model that learns richly structured multi-view plot representations, and show that they i) yield better micro-clusters than less structured representations; and ii) are interpretable, and thus useful for further literary analysis or labelling of the emerging micro-clusters.

1 Introduction
For the literature aficionado, the quest for the next novel to read can be daunting: the sheer number of novels of different styles, topics and genres is difficult to navigate. It is intuitively clear that readers select novels based on specific but potentially diverse and structured preferences (e.g., they might prefer novels of a particular theme (small-town romance), mood (dark) or based on character types (grumpy boss), character relations (love, enmity) and their development). These preferences also manifest in the organization of online book stores or recommendation platforms.1 For example, the Amazon book catalog contains semantic tags provided by experts (publishers), including labels of character types (pirates) or theme (secret baby romance) to aid focused search for novels of interest. Although these tags are already fairly granular, many cover large sets of novels (e.g., the tag secret baby romance covers almost 4,000 novels), limiting their utility for exhaustive exploration and call for even finer grained micro-groupings. Can we instead automatically induce fine-grained novel clusters in an unsupervised, data-driven way?

We propose a framework to learn structured, interpretable book representations that capture different aspects of the plot, and verify that such representations are rich enough to support downstream tasks like generating interpretable book groupings. A real-world application of this work is content-based book recommendation based on diverse and interpretable book characteristics. Content-based recommendation has been criticized by the limited complexity of typically employed features (limited content analysis; Lops et al. (2011); Adomavicius and Tuzhilin (2005)). This work addresses this problem by inducing complex, structured and interpretable representations. Our contributions are two-fold.

First, assuming that richly structured book tags call for rich content representations (which expert taggers arguably possess), we describe a deep unsupervised model for learning multi-view representations of novels. We use the term view to refer to specific types of plot characteristics (e.g., pertaining to events, characters or mood), and multi-view to refer to combinations of these views. We use multi-view book representations to construct meaningful and locally coherent neighbourhoods in model space, which we will refer to as micro-clusters. To this end, we extend a recent autoencoder model (Iyyer et al., 2016) to learn multi-view representations of books. Our model...
encodes properties of characters (view $v_1$), relations between characters (view $v_2$), and their respective trajectories over the plot. View-specific encodings are learnt in an unsupervised way from raw text as separate sets of word clusters which are jointly optimized to encode relevant and distinct information. These properties are crucial for applications such as book recommendation, because they allow to i) explain why particular books are similar based on the inferred latent structure and ii) find similarities based on important and distinct aspects of a novel (character types or interactions). Our framework of unsupervised multi-view learning is very flexible and can straightforwardly be applied to learn arbitrary kinds and numbers of views from raw text.

Secondly, we propose an empirical evaluation framework. Before we can use models to extend existing categories as discussed above, it must be shown that the representations capture existing associations. To this end, we investigate whether micro-clusters derived from induced representations resemble reference clusters defined as groups of novels sharing tags in the Amazon catalog. While automatic induction of plot representations has attracted considerable attention (see Jockers (2013)), evaluation has remained largely qualitative and intrinsic. To the best of our knowledge, we are the first to investigate the utility of automatically induced plot representations on an extrinsic task at scale. We evaluate micro-clusters as local neighbourhoods in model space containing 10,000 novels under 50 reference tags.

We show that rich multi-view representations produce better micro-clusters compared to competitive but simpler models, and that interpretability of the learnt representations is not compromised despite the more complex objective. We also qualitatively demonstrate that high-quality micro-clusters emerge from a smaller, more homogeneous data set of Gutenberg novels.

2 Related Work

Automatically learning representations of book plots, as structured summaries of their content, has attracted much attention (cf. Jockers (2013) for a review). Unsupervised models have been proposed which, given raw text, extract prototypical event structure (McIntyre and Lapata, 2010; Chambers and Jurafsky, 2009), prototypical characters (Bamman et al., 2013, 2014; Elsner, 2012) and their social networks (Elson et al., 2010).

Other work focused on the dynamics of a plot, learning trajectories of relations between two characters (Iyyer et al., 2016; Chaturvedi et al., 2017). Iyyer et al. (2016) combine dictionary learning (Olshausen and Field, 1997) with deep recurrent autoencoders to learn interpretable character relationship descriptors. They show that their deep model learns better representations than conceptually similar topic models (Gruber et al., 2007; Chang et al., 2009). Here, we extend the model of Iyyer et al. (2016) to simultaneously induce multiple views on the plot.

Methodologically, our work falls into the class of multi-view learning, and we propose a novel formulation of the model objective which encourages encoding of distinct information in the views. Our objective function is inspired by prior work in multi-task learning and deep domain adaptation for classification (Ganin and Lempitsky, 2015; Ganin et al., 2016). They train neural networks to simultaneously learn classifiers which are accurate on their target task and are agnostic about feature fluctuation pertaining to domain shift. We adapt this idea to unsupervised models with a reconstruction objective and learn multi-view representations which efficiently encode the input data and, at the same time, learn to only encode information relevant for the particular view.

Evaluating induced plot representations is notoriously difficult. Most evaluation has resorted to manual inspection, or crowd-sourced human judgments of the coherence and interpretability of the representations (Iyyer et al., 2016; Chaturvedi et al., 2017). While such evaluations demonstrated that the induced representations are qualitatively valuable, it is not clear whether they are rich and general enough to be used for downstream tasks and applications. Others have used automatically created gold-standards of re-occurring character names across scripts (‘gang member’) (Bamman et al., 2013), prototypical plot templates (tropes, e.g., ‘corrupt corporate executive’) or manually created gold-standards of character types (Vala et al., 2016) or their relations (Massey et al., 2015; Chaturvedi et al., 2017) to automatically measure the intrinsic value of learnt representations. Here,

---

2 We argue that both characters, and their relations evolve throughout the plot: Heroes pick up new attitudes or skills, and utilize those to different extents; relations change and develop over time (hate to love, friendship to enmity and back).

3 https://www.gutenberg.org/
we investigate how these results extend to extrinsic tasks, and use structured plot representations for the task of inducing micro-clusters of novels.

Elsner (2012) depart from the above pattern, suggesting an extrinsic, albeit artificial, evaluation paradigm. Approaching plot understanding from the angle of its utility for summarization, they use kernel methods to learn character-centric plot representations. They evaluate their trained models on their ability to differentiate between real and artificially distorted novels (e.g., with shuffled chapters). While this evaluation is extrinsic and quantitative, it leverages artificial data and it is not clear how the results extend to real-world summaries.

Language features were previously used in content-based book recommendation e.g., as bags-of-words (Mooney and Roy, 1999) or semantic frames (Clercq et al., 2014). Both works use structured databases and plot summaries rather than the raw book text. Other work used topic models to augment a recommender system of scientific articles (Wang and Blei, 2011). Similar to our work, these works emphasize the added value of interpretable representations and recommendations, however, they do not leverage the raw content of entire novels and the richness of information encoded in those.

3 Multi-View Novel Representations

We first provide an intuitive description of Relationship Modeling Networks (RMN; Iyyer et al. 2016), and our extension (henceforth MVPPlot), which jointly induces temporally aware multi-view representations of novel plots. Afterwards we describe the MVPPlot model in technical detail.

3.1 Intuition

Iyyer et al. (2016) introduce the RMN, an unsupervised model which learns interpretable plot representations in terms of types of relations between pairs of book characters, and their development over time. Given a book and a character pair, the model learns relation types as word clusters (not unlike topics in a topic model (Blei et al., 2003)) from local contexts mentioning both characters. In addition the RMN learns for each character pair how these relations vary over time as a trajectory of relations. Methodologically, the RMN combines a deep recurrent autoencoder with dictionary learning, where terms in the dictionary are relationship descriptors. The RMN learns to efficiently encode local text spans as a linear combination of these relation descriptors.

We extend RMNs to induce temporally aware multi-view representations of novel plots. Multiple interpretable views are induced jointly within one process in an unsupervised way. The core of our model closely corresponds to the structure of the RMN (as technically described in Section 3.2). However, we provide the model with distinct types of informative input for each view, and, reformulate the objective in a way that jointly optimizes parameterizations of all views to encode distinct information (cf., Section 3.3).

Our MVPPlot model learns two views: properties associated with individual characters ($v_1$), relations between character pairs ($v_2$, as in the RMN) and their respective development over the course of the plot (examples of descriptors learnt by MVPPlot for both views are shown in Table 1). Our modeling framework, however, is very general in the sense that any number and type of views can be learnt jointly as long as input with relevant signals can be provided for each view. For example, we could naturally extend the model described here with a ‘plot’ view to capture properties of the story which are not related to any character.

<table>
<thead>
<tr>
<th>View</th>
<th>Descriptor</th>
</tr>
</thead>
<tbody>
<tr>
<td>$v_1$</td>
<td>laugh, scream, laughing, yell, joke, cringe, disgrace, embarrassment, hate, cursing</td>
</tr>
<tr>
<td>$v_1$</td>
<td>snug, fleece, warm, comfortable, wet blanket, flannel, cozy, comfort, roamly</td>
</tr>
<tr>
<td>$v_1$</td>
<td>excellency, mademoiselle, monsieur, majesty, duchess, empress, madame, countess, madam</td>
</tr>
<tr>
<td>$v_2$</td>
<td>love, loving, lovely, dear, sweetest, dearest, thank, darling, congratulation, hello, associate, assistant, senior, chairman, executive</td>
</tr>
<tr>
<td>$v_2$</td>
<td>leadership, vice, director, liaison, vice-president</td>
</tr>
</tbody>
</table>

Table 1: Example property ($v_1$) and relation ($v_2$) descriptors induced by MVPPlot on the Gutenberg corpus, as their nearest neighbours in GloVe space.
Figure 1: Visualization of the MVPlot model.

linearly ordered sequences of text spans \( s^t \) at time \( t=\{1...T\} \) in which character \( c \) is mentioned, but no other character,

\[
S^{c,b}_{v1}=\{s^1, s^2, ..., s^T\} \text{ s.th. } \forall t : c \in s^t
\]

Similarly, \( S^{c_1,c_2,b}_{v2} \), given a book \( b \) and a pair of characters \( c_1 \) and \( c_2 \), contains linearly ordered text spans which mention both \( c_1 \) and \( c_2 \), but no other character,

\[
S^{c_1,c_2,b}_{v2}=\{s^1, s^2, ..., s^T\} \text{ s.th. } \forall t : c_1 \in s^t, c_2 \in s^t.
\]

The rest of the input preparation follows Iyyer et al. (2016) as follows. We map text spans into word embedding space, by mapping each word \( w \) to its 300-dimensional GloVe embedding \( e^w \) (Pennington et al., 2014) pre-trained on CommonCrawl, and averaging the word embeddings,

\[
e^t = \frac{1}{|s^t|} \sum_{w \in s^t} e^w.
\]  

We provide MVPlot with a trainable matrix \( B \) of dimensions \( b \times n \), where \( b \) is the number of books in our data set, and each row \( e^b \) is an \( n \)-dimensional book embedding, encoding background information (e.g., about its general setting or style) which is relevant to neither view of MVPlot. Finally the span embedding and the corresponding book embedding are concatenated, and passed through a ReLu non-linearity (cf., Figure 1, bottom),

\[
h^t = ReLu(W_{h}[e^t; e^b]).
\]  

Model architecture MVPlot uses the architecture of the RMN autoencoder, but replicates it for each input view, \( v_1 \) and \( v_2 \) (cf., Figure 1, center). Each part will induce an encoding of view-specific information. The feed-forward pass, described below, is identical for both parts, however, the loss and backpropagation will differ (cf., Section 3.3).

We describe the feed-forward pass for \( v_2 \), noting that it works analogously for \( v_1 \). The latent input representation \( h^t \) (eqn (2)) is passed through a softmax layer which returns a weight vector over descriptors, \( d^t_{v2} = softmax(W^d_{v2}[h^t]) \). Descriptors are rows in the \( k \times d \)-dimensional descriptor matrix \( R_{v2} \), with each row \( k \) corresponding to one \( d \)-dimensional descriptor (similar to a topic in a topic model). The input \( e^t \) is reconstructed through the dot product of \( d^t_{v2} \) and the descriptor matrix \( R_{v2} \),

\[
r^t = d^t_{v2} R_{v2}.
\]

Like in the original RMN, we want to capture the temporal development of character relations or properties. Intuitively, we assume that the relations between (or properties of) characters at time \( t \) depend on their relations (or properties) at time \( t-1 \). As in the RMN, we factor the descriptor weights of the previous time step \( d^{t-1} \) into the representation at time \( t \), such that

\[
d^t_{v2} = \alpha softmax(W^u_{v2}[h_t; d^{t-1}_{v2}]) + (1-\alpha)d^{t-1}\]

Output First, the model induces property descriptors (rows in \( R_{v1} \)) and the relationship descriptors (rows in \( R_{v2} \)). Both sets of descriptors are optimized to reconstruct model input in GloVe embedding space (cf., Section 3.3 for details). They consequently themselves live in GloVe word embedding space, and can be visualized through their nearest neighbours in this space. In addition, for each book \( b \), character \( e^b \) and character pair \( \{c_1, c_2\} \), sequences of weight vectors over relations

\[
T^{c_1,c_2,b}_{v2} = d^1_{v2}...d^T_{v2},
\]

and over properties

\[
T^{c,b}_{v1} = d^1_{v1}...d^T_{v1}
\]

are induced, which encode their trajectory of relations and properties, respectively. We will utilize these trajectories for inducing micro-clusters of novels (Section 6.1).
3.3 The Multi-View Loss

We formulate our loss as a Hinge loss within the contrastive max-margin framework. Our objective is to learn parameters for each view $v \in \{v_1, v_2\}$ which efficiently encode view-specific input in a low-dimensional space from which the original input can be re-constructed with high accuracy. In addition, we want to learn view-specific parameters which encode distinct information such that when utilized together, they provide an improved embedding of the data. Intuitively, we achieve this by discouraging parameters of view $v_1$ from accurately reconstructing input spans from view $v_2$, and vice versa.

Our loss combines these two objectives as follows. The first part of the loss corresponds to the loss of the RMN. We use negative sampling to induce parameters for each view which reconstruct their respective view-specific input well. Formally, assuming model input from view $v_1$, $e_{v_1}^t$, we construct a set of 10 ‘negative inputs’ $\{e_{v_1}^{n_1},...,e_{v_1}^{n_{10}}\}$ which are sampled at random from the $v_1$ input corpus. We want to learn parameters encoding view $v_1$ to reconstruct the input such that the inner product between the true input $e_{v_1}^t$ and its reconstruction $r_{v_1}^t$ is higher than the inner product between $r_{v_1}^t$ and any of the negative samples $e_{v_1}^{n_i}$ by a margin of at least 1,

$$J(\theta) = \sum_i \sum_i \max(0, 1 - r_{v_1}^t e_{v_1}^t + r_{v_1}^t e_{v_1}^{n_i}),$$

where $\theta$ refers to the set of all model parameters. We add an orthogonality-encouraging regularization term to this objective in order to obtain view-specific descriptors which are distant from each other (Hyvärinen and Oja, 2000),

$$J(\theta) = \sum_i \sum_i \max(0, 1 - r_{v_1}^t e_{v_1}^t + r_{v_1}^t e_{v_1}^{n_i}) + \lambda \|R_{v_1} R_{v_1}^T - I\|.$$  (6)

The loss is defined analogously for input of view $v_2$. Note that so far, the loss is defined in an entirely view-specific way, independent of the $v_2$ parameters (e.g., the $v_1$ loss in equation (6) is independent of the $v_2$ parameters).

We break this independence by adding a second term to our loss function, which ensures that view-specific parameters encode only relevant information. That is, we want $v_2$-specific parameters to only encode $v_2$-specific information, and vice versa. Assuming model input from view $v_1$, $e_{v_1}^t$, we learn parameters for to view $v_2$ that reconstruct the input poorly. Again, we use the max-margin framework, maximizing the margin between the (high) quality reconstruction of $e_{v_1}^t$ from $v_1$ parameters, $r_{v_1}^t$, and the (poor) quality of the reconstruction from $v_2$ parameters, $r_{v_2}^t$.

$$K(\theta) = \max(0, 1 - e_{v_1}^t r_{v_1} + e_{v_2}^t r_{v_2}).$$  (7)

The update is defined analogously, swapping $v_1$ and $v_2$ subscripts, when the true input stems from $v_2$. The full loss is defined as a weighted linear combination of its terms (eqns (6) and (7)),

$$L(\theta) = \beta L(\theta) + (1 - \beta) K(\theta).$$  (8)

4 Semantic Micro-Cluster Evaluation

MVPlot induces structured representations of a novel $b$ as relation trajectories between characters pairs in $b$, and property trajectories of individual characters in $b$. Are those representations rich and informative enough to produce meaningful and interpretable micro-clusters of novels? In Section 6.1 we evaluate the quality of such micro-clusters, i.e., local novel neighbourhoods in model space. We propose an objective and empirical evaluation employing expert-provided semantic novel tags in the Amazon catalog.

Novels listed in the Amazon catalog are tagged with respect to their genre (e.g., mystery, romance). They are further labelled with refinements pertaining to diverse information like character types or mood, which take different sets of values, depending on the genre, and are as such predestined as an objective reference for evaluating the diverse information captured by our model. Table 2 lists example tags for the refinement character type.

<table>
<thead>
<tr>
<th>Genre</th>
<th>Example Tags</th>
</tr>
</thead>
<tbody>
<tr>
<td>Mystery</td>
<td>British Detectives; FBI Agents; Female Protagonists;</td>
</tr>
<tr>
<td></td>
<td>Private Investigators</td>
</tr>
<tr>
<td>Romance</td>
<td>Cowboys; Criminals &amp; Outlaws; Doctors; Royalty &amp;</td>
</tr>
<tr>
<td></td>
<td>Aristocrats; Spies; Wealthy</td>
</tr>
<tr>
<td>SciFi</td>
<td>Aliens; Aliens; Clones; Corporations; Mutants;</td>
</tr>
<tr>
<td></td>
<td>Pirates; Psychics; Robots &amp; Androids</td>
</tr>
</tbody>
</table>

Table 2: Example tags from the Amazon book catalog for the refinement character type.
Table 3: The number of novels and property ($v_1$) relation ($v_2$) input sequences for the Gutenberg and the Amazon corpus.

<table>
<thead>
<tr>
<th></th>
<th># novels</th>
<th>$v_1$ sequences</th>
<th>$v_2$ sequences</th>
</tr>
</thead>
<tbody>
<tr>
<td>Gutenberg</td>
<td>3,500</td>
<td>45,182</td>
<td>60,493</td>
</tr>
<tr>
<td>Amazon</td>
<td>10,000</td>
<td>91,511</td>
<td>70,156</td>
</tr>
</tbody>
</table>

We selected a set of 50 representative tags from the Amazon catalog and did not tune this set for our evaluation. The full tag set is included in the supplementary material.

Note that while this scheme provides an empirical way of evaluating plot representations, it may not capture their full potential: our models are not explicitly tuned towards producing micro-clusters which are coherent with respect to our gold-standard tags, and may encode additional structure which is not probed in this evaluation. That said, we consider this evaluation as a good procedure to evaluate the relative quality of different models in the sense that a better model should produce micro-clusters that better correspond to reference clusters derived from gold-standard tags.

5 Data

We evaluate our model on two data sets. First, we create a diverse data set by sampling 10,000 digital novels under our 50 gold-standard tags (cf., Section 4) of the Amazon catalog (Amazon). Our second data set consists of 3,500 novels from Project Gutenberg, a large digital collection of freely available novels consisting primarily of classic literature (Gutenberg). The Amazon novels are already labelled with genre and refinement tags, such that evaluation using our gold-standard is straightforward. While Gutenberg novels come with the advantage of being freely available, they are unlabelled, and not fully covered by our 50 gold-standard tags. We therefore restrict our quantitative analysis to the Amazon data set. However, we also report qualitative results on the Gutenberg corpus, demonstrating that our model induces meaningful novel representations for corpora of varying size and diversity.

Both data sets were pre-processed with the BookNLP pipeline (Bamman et al., 2014) for coreference resolution of character mentions. We filtered stop-words and low-frequency words by discarding the 500 most frequent words and those which occur in less than 100 novels, and discarded novels less than 100 sentences long or containing fewer than 5 characters from our data set.

We created view-specific input corpora as follows: (1) a relation corpus of chronologically ordered sequences of text spans of 20 words for character pairs $\{c_1, c_2\}$ in a book $b$, $S_{v_2}^{c_1,c_2,b}$, which mention only $c_1$ and $c_2$ with a margin of 10 words for the Amazon corpus (1 word for the smaller Gutenberg corpus) but no other character; and (2) a property corpus of chronologically ordered sequences of 20 word text spans for individual characters $c$ in book $b$, $S_{v_2}^{c_1,b}$, which mention only $c$, using the same margins as above.

We keep only sequences of length $n$ time steps s.th., $5 \leq n \leq 250$. We only keep pair sequences if we also obtain sequences for each individual character confirming to the above criteria. Table 3 summarizes statistics on our input corpora.

6 Evaluation

Section 6.1 quantitatively evaluates the quality of local neighbourhoods in model space induced from the Amazon corpus against our proposed gold-standard. Section 6.2 evaluates the quality of the induced descriptors from both the Amazon and Gutenberg corpus both through crowd sourcing and illustrative examples.

Models We set the MVPlot performance into perspective comparing it against the RMN.\(^6\) MV-Plot induces both character properties and relations, and is trained on both the relation-view and property-view input, while the RMN only induces pair relationships and can only utilize relation-view input. In addition, we report a frequency baseline, which is trained on both property and relation-view input. We concatenate all input spans of a given view for a particular novel; construct its term frequency vector and use cosine similarity to compute the nearest neighbours to each novel.

Parameter settings Across all experiments and corpus-specific models, we set $\beta=0.99$ for MV-Plot, and for both MVPlot and RMN we set $\alpha=0.5$, $\lambda=10^{-5}$, $k=50$.\(^7\) We train both RMN and MVPPlot for 15 epochs using SGD and ADAM (Kingma and Ba, 2014).\(^8\)

---

\(^6\)We do not compare against topic model baselines because they were outperformed by RMN (Iyyer et al., 2016).

\(^7\)Parameters were tuned on a small subset of books used in the nearest neighbourhood evaluation (Section 6.1).

\(^8\)Our implementation builds on the available RMN code https://github.com/miyyer/rmn.
6.1 Nearest Neighbours Evaluation

We evaluate local neighbourhoods in model space using the 500 most popular novels by their number of Amazon reviews as reference novels from the Amazon corpus. For each reference novel we compute the 10 nearest neighbours as described below. We measure neighbourhood quality using the gold-standard tags from Section 4, regarding neighbours as relevant if at least one tag is shared with the reference novel. We report precision at rank 10 (P@10) and mean average precision (MAP).

Method MVPlot represents a book \( b \) in terms of trajectories of weight vectors over relation descriptors \( T^b \) and property descriptors \( T^b_0 \). RMN only learns relation descriptors and their trajectories. For both models, we map each induced trajectory for book \( b \) to a fixed-sized \( k \)-dimensional vector representation by averaging the time-specific weight vectors, for example for a \( v_2 \) trajectory,

\[
T^{v_2}_{c_1, c_2, b} = \frac{1}{|T^{v_2}_{c_1, c_2, b}|} \sum_{t \in T^{v_2}_{c_1, c_2, b}} d^{v_2}_t, \tag{9}
\]

and equivalently for \( v_1 \) trajectories, \( T^{v_1}_{c_1, b} \).

We compute the similarity between two books \( \{b_r, b_c\} \) as follows. We align the \( v_2 \) trajectory for each character pair \( \{c_1, c_2\} \) in \( b_r \), \( T^{c_1, c_2, c_r} \), to its closest neighbouring character pair vector in \( b_c \), \( T^{c_1, c_2, c_c} \), by Euclidean distance, and compute the overall book similarity in terms of character relations between \( b_r \) and \( b_c \) as the average over all distances.

\[
sim^{b_r, b_c}_{v_2} = \frac{1}{|T^{v_2}_{c_1, c_2, b_r}|} \sum_{T^{c_1, c_2, c_r} \in T^{c_1, c_2, b_r}} \arg \min_{T^{c_1, c_2, c_c} \in T^{c_1, c_2, b_c}} \text{dist}(T, T'). \tag{10}
\]

We obtain \( \sim^{b_r, b_c}_{v_2} \) in an analogous process. For cosine and MVPlot we obtain a final, multi-view similarity by averaging similarity scores obtained in each view’s space,

\[
sim^{b_r, b_c}_{both} = \frac{1}{2} \left( \sim^{b_r, b_c}_{v_1} + \sim^{b_r, b_c}_{v_2} \right). \tag{11}
\]

For RMN we compute similarity only in character relation space.

Results Table 4 presents micro-cluster quality in terms of precision@10 and MAP. The full MVPlot model statistically significantly outperforms all other models.\footnote{Also, intra-view comparisons except for MVPlot \( v_1 \) and cosine \( v_1 \) are statistically significant.} The same pattern emerges when comparing models with the same underlying views: MVPlot \( v_2 \) outperforms both cosine \( v_2 \) and RMN \( v_2 \) (similarly for MVPlot \( v_1 \) and cosine \( v_1 \)), indicating that the MVPlot character relation representations are most informative for micro-cluster induction.

In order to shed light on the contribution of individual model components, we compare the full MVPlot model (\( \text{both} \)) to model versions with access to only \( v_1 \) or \( v_2 \) (Table 4 bottom). Combining information from both views boosts performance compared to the single-view versions. This confirms that MVPlot indeed encodes distinct and relevant information in the respective views.

While cosine is a strong baseline, its representations are not structured or interpretable. It consequently does not provide sufficient information for applications like book tagging or recommendation with respect to specific aspects or criteria. Similarly, RMN cannot learn representations of multiple, distinct views of the plot.

Advancing our understanding of the information encoded in the individual views of MVPlot, we took a closer look at the refinement tags for which the single view MVPlot model (\( v_1 \)) has the clearest advantage over the pair view MVPlot model (\( v_2 \)) and vice-versa. We computed tag-wise F1-scores for the two MVPlot variants. Table 5 lists the book tags for which the scores of the two views diverge the most.

In terms of types of refinements, view \( v_2 \) suffers most for predicting book categories, or topical tags (‘sports’, ‘second changes’), while view \( v_1 \) is particularly deficient for predicting character types. While this seems counterintuitive we hypothesize that character types are to a large extent defined by their interactions with, or relations to, other char-

<table>
<thead>
<tr>
<th>Model</th>
<th>View</th>
<th>( P@10 )</th>
<th>MAP</th>
</tr>
</thead>
<tbody>
<tr>
<td>( v_1 )</td>
<td>cosine</td>
<td>0.516 †‡</td>
<td>0.392 †‡</td>
</tr>
<tr>
<td>( v_2 )</td>
<td>cosine</td>
<td>0.468 †‡</td>
<td>0.339 †‡</td>
</tr>
<tr>
<td>both</td>
<td>cosine</td>
<td>0.512 †‡</td>
<td>0.390 †‡</td>
</tr>
<tr>
<td>RMN</td>
<td>( v_2 )</td>
<td>0.479 †‡</td>
<td>0.347 †‡</td>
</tr>
<tr>
<td>( v_1 )</td>
<td>RMN</td>
<td>0.529 †‡</td>
<td>0.401 †‡</td>
</tr>
<tr>
<td>MVPlot</td>
<td>( v_2 )</td>
<td>0.496 †‡</td>
<td>0.367 †‡</td>
</tr>
<tr>
<td>both</td>
<td>MVPlot</td>
<td>0.546 0.421</td>
<td></td>
</tr>
</tbody>
</table>

Table 4: Micro-cluster quality results (Amazon corpus). Differences of cosine and RMN compared to the best MVPlot result are significant with \( p < 0.05 (†) \) or \( p < 0.01 (‡) \) (paired t-test).
Table 5: The tags (Tag) and their refinement types (RefType) for which MVPlot v1 most clearly outperforms MVPlot v2 (left) and vice versa (right) in terms of tag-specific F1-measure.

<table>
<thead>
<tr>
<th>Tag</th>
<th>RefType</th>
<th>Tag</th>
<th>RefType</th>
</tr>
</thead>
<tbody>
<tr>
<td>Robots &amp; Androids</td>
<td>Character</td>
<td>Hard SciFi</td>
<td>Category</td>
</tr>
<tr>
<td>Corporations</td>
<td>Character</td>
<td>Sports</td>
<td>Category</td>
</tr>
<tr>
<td>International</td>
<td>Theme</td>
<td>Horror</td>
<td>Theme</td>
</tr>
<tr>
<td>Aliens</td>
<td>Character</td>
<td>Second Chances</td>
<td>Theme</td>
</tr>
<tr>
<td>Cowboys</td>
<td>Character</td>
<td>Crime</td>
<td>Category</td>
</tr>
</tbody>
</table>

Figure 2: Results of descriptor interpretability. (% of descriptors marked as ‘relevant descriptors of relations’ by various proportions of annotators).

Figure 3 further illustrates this, displaying example local neighbourhoods of four reference novels (left) with their eight nearest neighbours ordered by proximity (left to right). The neighbourhoods are intuitively meaningful, and particularly impressive bearing in mind that the full model space contains 3,500 novels. While most neighbourhoods are dominated by novels of the same author, some exceptions emerge. Row two, for example, contains novels by Thomas Hardy and Charles Dickens who both are known for biographical 17th century novels focusing on class and social changes.

7 Conclusions

Content-based micro-clustering of novels is a complex but interesting task. In order to eventually augment the diverse associations humans have, models must be able to pick up rich and structured signals from raw text. This paper presented a deep recurrent autoencoder which learns multi-view representations of plots, and introduced a principled evaluation framework using clusters based on expert-provided book tags. Our evaluation showed that rich multi-view representations are better suited to recover such reference clusters compared to each individual view, as well as compared to simpler, but competitive models which induce less structured representations. Our view-specific representations are interpretable which allows to analyse and explain the emerging
Figure 3: Nearest neighbours for four classic stories from the Gutenberg Corpus. Target novels on the left (with red border), and NNs are presented in the same row, ordered by their distance to the target novel.

micro-clusters, and might reveal previously unnoticed parallels between novels and may be useful for literary analysis or content-based recommendation. This is an exciting avenue for future work.

Our method is general and scalable both in terms of its input, utilizing raw text with only automatic pre-processing, and in terms of the number of distinct views it can learn. We described an objective function which triggers views to encode distinct information. In future work we plan to explore joint learning of more and different views.

Our approach relies strongly on the assumption that text spans mentioning two characters contain information about character relations, and that text spans mentioning one character contain information about the character’s properties. While our results suggest that these assumptions are valid, they are arguably crude. In the future we plan to define more targeted input, e.g., by using semantic and syntactic information from dependency parses.

In this work we induced dual-view representations of book content, however, we emphasize that the proposed method is very general. The number and kinds of views, as well as underlying data are in no way constrained, as long as relevant view-specific input can be defined. In the context of novel representation it would be interesting to introduce additional views, for example one that captures the mood of a novel. Another interesting avenue for future work would be to apply the framework to questions arising in the digital humanities, e.g., to extract different views from news articles.

The presented model and evaluation are designed with the objective to detect a different kinds of similarity between novels, with the ultimate goal to enrich human-provided genres and tags. We described a first step in this direction, verifying that the learnt information is meaningful and can reproduce human-created semantic book tags. Expert book tags exist for a wide variety of information (mood, theme, characters), and provide a rich evaluation environment for learnt representations. We invite the community to join us in exploring the full space of opportunities and evaluating induced representations holistically in the future.

Acknowledgements

We would like to thank Alex Klementiev, Kevin Small, Joon Hao Chuah and Mohammad Kanso for their valuable insights, feedback and technical help on the work presented in this paper. We also thank the anonymous reviewers for their valuable feedback and suggestions.
References


