**Aligning Books and Movies: Towards Story-Like Visual Explanations by Watching Movies and Reading Books**

The MIT Faculty has made this article openly available. **Please share** how this access benefits you. Your story matters.

<table>
<thead>
<tr>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>As Published</td>
<td><a href="http://dx.doi.org/10.1109/ICCV.2015.11">http://dx.doi.org/10.1109/ICCV.2015.11</a></td>
</tr>
<tr>
<td>Publisher</td>
<td>Institute of Electrical and Electronics Engineers (IEEE)</td>
</tr>
<tr>
<td>Version</td>
<td>Original manuscript</td>
</tr>
<tr>
<td>Accessed</td>
<td>Thu Dec 06 09:20:08 EST 2018</td>
</tr>
<tr>
<td>Citable Link</td>
<td><a href="http://hdl.handle.net/1721.1/112996">http://hdl.handle.net/1721.1/112996</a></td>
</tr>
<tr>
<td>Terms of Use</td>
<td>Creative Commons Attribution-Noncommercial-Share Alike</td>
</tr>
<tr>
<td>Detailed Terms</td>
<td><a href="http://creativecommons.org/licenses/by-nc-sa/4.0/">http://creativecommons.org/licenses/by-nc-sa/4.0/</a></td>
</tr>
</tbody>
</table>
Aligning Books and Movies: Towards Story-like Visual Explanations by Watching Movies and Reading Books

Yukun Zhu *,1 Ryan Kiros *,1 Richard Zemel 1 Ruslan Salakhutdinov 1
Raquel Urtasun 1 Antonio Torralba 2 Sanja Fidler 1
1University of Toronto 2Massachusetts Institute of Technology
{yukun,rkiros,zemel,rsalakhu,urtasun,fidler}@cs.toronto.edu, torralba@csail.mit.edu

Abstract

Books are a rich source of both fine-grained information, how a character, an object or a scene looks like, as well as high-level semantics, what someone is thinking, feeling and how these states evolve through a story. This paper aims to align books to their movie releases in order to provide rich descriptive explanations for visual content that go semantically far beyond the captions available in current datasets.

To align movies and books we exploit a neural sentence embedding that is trained in an unsupervised way from a large corpus of books, as well as a video-text neural embedding for computing similarities between movie clips and sentences in the book. We propose a context-aware CNN to combine information from multiple sources. We demonstrate good quantitative performance for movie/book alignment and show several qualitative examples that showcase the diversity of tasks our model can be used for.

1. Introduction

A truly intelligent machine needs to not only parse the surrounding 3D environment, but also understand why people take certain actions, what they will do next, what they could possibly be thinking, and even try to empathize with them. In this quest, language will play a crucial role in grounding visual information to high-level semantic concepts. Only a few words in a sentence may convey really rich semantic information. Language also represents a natural means of interaction between a naive user and our vision algorithms, which is particularly important for applications such as social robotics or assistive driving.

Combining images or videos with language has gotten significant attention in the past year, partly due to the creation of CoCo [18]. Microsoft’s large-scale captioned image dataset. The field has tackled a diverse set of tasks such as captioning [13, 11, 36, 35, 21], alignment [11, 15, 34], Q&A [20, 19], visual model learning from textual descriptions [8, 26], and semantic visual search with natural multi-sentence queries [17].

Denotes equal contribution

Figure 1: Shot from the movie Gone Girl, along with the subtitle, aligned with the book. We reason about the visual and dialog (text) alignment between the movie and a book.

Books provide us with very rich, descriptive text that conveys both fine-grained visual details (how people or scenes look like) as well as high-level semantics (what people think and feel, and how their states evolve through a story). This source of knowledge, however, does not come with associated visual information that would enable us to ground it with descriptions. Grounding descriptions in books to vision would allow us to get textual explanations or stories behind visual information rather than simplistic captions available in current datasets. It can also provide us with extremely large amount of data (with tens of thousands books available online).

In this paper, we exploit the fact that many books have been turned into movies. Books and their movie releases have a lot of common knowledge as well as they are complementary in many ways. For instance, books provide detailed descriptions about the intentions and mental states of the characters, while movies are better at capturing visual aspects of the settings.

The first challenge we need to address, and the focus of this paper, is to align books with their movie releases in order to obtain rich descriptions for the visual content. We aim to align the two sources with two types of information: visual, where the goal is to link a movie shot to a book paragraph, and dialog, where we want to find correspondences between sentences in the movie’s subtitle and sentences in the book. We formulate the problem of movie/book alignment as finding correspondences between shots in the movie as well as dialog sentences in the subtitles and sentences in the book (Fig. 1). We introduce a novel sentence similarity measure based on a neural sen-
ference embedding trained on millions of sentences from a large corpus of books. On the visual side, we extend the
neural image-sentence embeddings to the video domain and train the model on DVS descriptions of movie clips. Our
approach combines different similarity measures and takes into account contextual information contained in the nearby
shots and book sentences. Our final alignment model is formulated as an energy minimization problem that encourages
the alignment to follow a similar timeline. To evaluate the
book-movie alignment model we collected a dataset with
11 movie/book pairs annotated with 2,070 shot-to-sentence
correspondences. We demonstrate good quantitative perfor-
ance and show several qualitative examples that showcase
the diversity of tasks our model can be used for.

The alignment model can have multiple applications.
Imagine an app which allows the user to browse the book
as the scenes unroll in the movie: perhaps its ending or act-
ing are ambiguous, and one would like to query the book
for answers. Vice-versa, while reading the book one might
want to switch from text to video, particularly for the juicy
scenes. We also show other applications of learning from
movies and books such as book retrieval (finding the book
that goes with a movie and finding other similar books), and
captioning CoCo images with story-like descriptions.

2. Related Work

Most effort in the domain of vision and language has
been devoted to the problem of image captioning. Older
work made use of fixed visual representations and translated
them into textual descriptions [6, 16]. Recently, several
approaches based on RNNs emerged, generating captions
via a learned joint image-text embedding [13, 11, 36, 21].
These approaches have also been extended to generate de-
scriptions of short video clips [35]. In [24], the authors go
beyond describing what is happening in an image and pro-
vide explanations about why something is happening.

For text-to-image alignment, [15, 7] find correspon-
dences between nouns and pronouns in a caption and visual
objects using several visual and textual potentials. Lin et
al. [17] does so for videos. In [11], the authors use RNN
embeddings to find the correspondences. [37] combines
neural embeddings with soft attention in order to align the
words to image regions.

Early work on movie-to-text alignment include dynamic
time warping for aligning movies to scripts with the help
of subtitles [5, 4]. Sankar et al. [28] further developed a
system which identified sets of visual and audio features to
align movies and scripts without making use of the subtitles.
Such alignment has been exploited to provide weak labels
for person naming tasks [5, 30, 25].

Closest to our work is [34], which aligns plot synopses to
shots in the TV series for story-based content retrieval. This
work adopts a similarity function between sentences in plot
synopses and shots based on person identities and keywords
in subtitles. Our work differs with theirs in several impor-
tant aspects. First, we tackle a more challenging problem of
movie/book alignment. Unlike plot synopsis, which closely
follow the storyline of movies, books are more verbose and
might vary in the storyline from their movie release. Fur-
thermore, we use learned neural embeddings to compute the
similarities rather than hand-designed similarity functions.

Parallel to our work, [33] aims to align scenes in movies
to chapters in the book. However, their approach operates
on a very coarse level (chapters), while ours does so on the
sentence/paragraph level. Their dataset thus evaluates on
90 scene-chapter correspondences, while our dataset draws
2,070 shot-to-sentences alignments. Furthermore, the ap-
proaches are inherently different. [33] matches the pres-
ence of characters in a scene to those in a chapter, as well
as uses hand-crafted similarity measures between sentences
in the subtitles and dialogs in the books, similarly to [34].

Rohrbach et al. [27] recently released the Movie De-
scription dataset which contains clips from movies, each
time-stamped with a sentence from DVS (Descriptive Video
Service). The dataset contains clips from over a 100 movies,
and provides a great resource for the captioning techniques.
Our effort here is to align movies with books in order to ob-
tain longer, richer and more high-level video descriptions.

We start by describing our new dataset, and then explain
our proposed approach.

3. The MovieBook and BookCorpus Datasets

We collected two large datasets, one for movie/book
alignment and one with a large number of books.

The MovieBook Dataset. Since no prior work or data ex-
ist on the problem of movie/book alignment, we collected a
new dataset with 11 movies along with the books on which
they were based on. For each movie we also have a sub-
title file, which we parse into a set of time-stamped sen-
tences. Note that no speaker information is provided in the
subtitles. We automatically parse each book into sentences,
paragraphs (based on indentation in the book), and chapters
(we assume a chapter title has indentation, starts on a new
page, and does not end with an end symbol).

Our annotators had the movie and a book opened side
by side. They were asked to iterate between browsing the
book and watching a few shots/scenes of the movie, and
trying to find correspondences between them. In particular,
they marked the exact time (in seconds) of correspondence
in the movie and the matching line number in the book file,
indicating the beginning of the matched sentence. On the
video side, we assume that the match spans across a shot (a
video unit with smooth camera motion). If the match was
longer in duration, the annotator also indicated the ending
time of the match. Similarly for the book, if more sentences
matched, the annotator indicated from which to which line a match occurred. Each alignment was also tagged, indicating whether it was a visual, dialogue, or an audio match. Note that even for dialogs, the movie and book versions are semantically similar but not exactly the same. Thus deciding on what defines a match or not is also somewhat subjective and may slightly vary across our annotators. Altogether, the annotators spent 90 hours labeling 11 movie/book pairs, locating 2,070 correspondences.

Table 1 presents our dataset, while Fig. 8 shows a few ground-truth alignments. One can see the complexity and diversity of the data: the number of sentences per book vary from 638 to 15,498, even though the movies are similar in duration. This indicates a huge diversity in descriptiveness across literature, and presents a challenge for matching. The sentences also vary in length, with the sentences in Brokeback Mountain being twice as long as those in The Road. The longest sentence in American Psycho has 422 words and spans over a page in the book.

Aligning movies with books is challenging even for humans, mostly due to the scale of the data. Each movie is on average 2h long and has 1,800 shots, while a book has on average 7,750 sentences. Books also have different styles of writing, formatting, different and challenging language, slang (going vs goin’, or even was vs ‘us), etc. As one can see from Table 1, finding visual matches turned out to be particularly challenging. This is because the visual descriptions in books can be either very short and hidden within longer paragraphs or even within a longer sentence, or very verbose – in which case they get obscured with the surrounding text – and are hard to spot. Of course, how close the movie follows the book is also up to the director, which can be seen through the number of alignments that our annotators found across different movie/books.

The BookCorpus Dataset. In order to train our sentence similarity model we collected a corpus of 11,038 books from the web. These are free books written by yet unpublished authors. We only included books that had more than 20K words in order to filter out perhaps noisier shorter stories. The dataset has books in 16 different genres, e.g., Romance (2,865 books), Fantasy (1,479), Science fiction (786), Teen (430), etc. Table 2 highlights the summary statistics of our book corpus.

4. Aligning Books and Movies

Our approach aims to align a movie with a book by exploiting visual information as well as dialogs. We take shots as video units and sentences from subtitles to represent dialogs. Our goal is to match these to the sentences in the book. We propose several measures to compute similarities between pairs of sentences as well as shots and sentences. We use our novel deep neural embedding trained on our large corpus of books to predict similarities between sentences. Note that an extended version of the sentence embedding is described in detail in [14] showing how to deal with million-word vocabularies, and demonstrating its performance on a large variety of NLP benchmarks. For comparing shots with sentences we extend the neural embedding of images and text [13] to operate in the video domain. We next develop a novel contextual alignment model that combines information from various similarity measures and a larger time-scale in order to make better local alignment predictions. Finally, we propose a simple pairwise Conditional Random Field (CRF) that smooths the alignments by encouraging them to follow a linear timeline, both in the video and book domain.

We first explain our sentence, followed by our joint video to text embedding. We next propose our contextual model that combines similarities and discuss CRF in more detail.

### 4.1. Skip-Thought Vectors

In order to score the similarity between two sentences, we exploit our architecture for learning unsupervised representations of text [14]. The model is loosely inspired by
the skip-gram [22] architecture for learning representations of words. In the word skip-gram model, a word \( w_i \) is chosen and must predict its surrounding context (e.g. \( w_{i+1} \) and \( w_{i-1} \) for a context window of size 1). Our model works in a similar way but at the sentence level. That is, given a sentence tuple \((s_{i-1}, s_i, s_{i+1})\) our model first encodes the sentence \( s_i \) into a fixed vector, then conditioned on this vector tries to reconstruct the sentences \( s_{i-1} \) and \( s_{i+1} \), as shown in Fig. 2. The motivation for this architecture is inspired by the distributional hypothesis: sentences that have similar surrounding context are likely to be both semantically and syntactically similar. Thus, two sentences that have similar syntax and semantics are likely to be encoded to a similar vector. Once the model is trained, we can map any sentence through the encoder to obtain vector representations, then score their similarity through an inner product.

The learning signal of the model depends on having contiguous text, where sentences follow one another in sequence. A natural corpus for training our model is thus a large collection of books. Given the size and diversity of genres, our BookCorpus allows us to learn very general representations of text. For instance, Table 3 illustrates the nearest neighbours of query sentences, taken from held out books that the model was not trained on. These qualitative results demonstrate that our intuition is correct, with resulting nearest neighbours corresponds largely to syntactically and semantically similar sentences. Note that the sentence embedding is general and can be applied to other domains not considered in this paper, which is explored in [14].

To construct an encoder, we use a recurrent neural network, inspired by the success of encoder-decoder models for neural machine translation [10, 2, 1, 31]. Two kinds of activation functions have recently gained traction: long short-term memory (LSTM) [9] and the gated recurrent unit (GRU) [3]. Both types of activation successfully solve the vanishing gradient problem, through the use of gates to control the flow of information. The LSTM unit explicitly employs a cell that acts as a carousel with an identity weight. The flow of information through a cell is controlled by input, output and forget gates which control what goes into a cell, what leaves a cell and whether to reset the contents of the cell. The GRU does not use a cell but employs two gates: an update and a reset gate. In a GRU, the hidden state is a linear combination of the previous hidden state and the proposed hidden state, where the combination weights are controlled by the update gate. GRUs have been shown to perform just as well as LSTM on several sequence prediction tasks [3] while being simpler. Thus, we use GRU as the activation function for our encoder and decoder RNNs.

Suppose that we are given a sentence tuple \((s_{i-1}, s_i, s_{i+1})\), and let \( w_{i} \) denote the \( t \)-th word for \( s_{i} \) and let \( x_{i} \) be its word embedding. We break the model description into three parts: the encoder, decoder and objective function.

**Encoder.** Let \( w_{i}^{1}, \ldots, w_{i}^{N} \) denote words in sentence \( s_{i} \) with \( N \) the number of words in the sentence. The encoder produces a hidden state \( h_{i}^{t} \) at each time step which forms the representation of the sequence \( w_{i}^{1}, \ldots, w_{i}^{t} \). Thus, the hidden state \( h_{i}^{N} \) is the representation of the whole sentence. The GRU produces the next hidden state as a linear combination of the previous hidden state and the proposed state update (we drop subscript \( i \)):

\[
\begin{align*}
\hat{h}_{i}^{t} & = (1 - z_{i}^{t}) \odot h_{i}^{t-1} + z_{i}^{t} \odot \tilde{h}_{i}^{t} \\
\end{align*}
\]

where \( \tilde{h}_{i}^{t} \) is the proposed state update at time \( t \), \( z_{i}^{t} \) is the update gate and \( \odot \) denotes a component-wise product. The update gate takes values between zero and one. In the extreme cases, if the update gate is the vector of ones, the previous hidden state is completely forgotten and \( h_{i}^{t} = \tilde{h}_{i}^{t} \). Alternatively, if the update gate is the zero vector, then the...
hidden state from the previous time step is simply copied over, that is \( h^t = h^{t-1} \). The update gate is computed as
\[
z^t = \sigma(W_z x^t + U_z h^{t-1})
\]
where \( W_z \) and \( U_z \) are the update gate parameters. The proposed state update is given by
\[
h^t = \tanh(W x^t + U(r^t \circ h^{t-1}))
\]
where \( r_t \) is the reset gate, which is computed as
\[
r^t = \sigma(W_r x^t + U_r h^{t-1})
\]
If the reset gate is the zero vector, then the proposed state update is computed only as a function of the current word. Thus after iterating this equation sequence for each word, we obtain a sentence vector \( h^T_s = h_t \) for sentence \( s_t \).

**Decoder.** The decoder computation is analogous to the encoder, except that the computation is conditioned on the sentence vector \( h_t \). Two separate decoders are used, one for the previous sentence \( s_{i-1} \) and one for the next sentence \( s_{i+1} \). These decoders use different parameters to compute their hidden states but both share the same vocabulary matrix \( V \) that takes a hidden state and computes a distribution over words. Thus, the decoders are analogous to an RNN language model but conditioned on the encoder sequence. Alternatively, in the context of image caption generation, the encoded sentence \( h_t \) plays a similar role as the image.

We describe the decoder for the next sentence \( s_{i+1} \) (computation for \( s_{i-1} \) is identical). Let \( h_{i+1} \) denote the hidden state of the decoder at time \( t \). The update and reset gates for the decoder are given as follows (we drop \( i+1 \)):
\[
z^t = \sigma(W_z x^t + U_z h^{t-1} + C_z h_t)
\]
\[
r^t = \sigma(W_r x^t + U_r h^{t-1} + C_r h_t)
\]
the hidden state \( h_{i+1} \) is then computed as:
\[
h^t = \tanh(W d x^{t-1} + U d (r^t \circ h^{t-1}) + C h_t)
\]
\[
h_{i+1} = (1 - z^t) \circ h^{t-1} + z^t \circ h^t
\]
Given \( h_{i+1} \), the probability of word \( w_{i+1} \) given the previous \( t-1 \) words and the encoder vector is
\[
P(w_{i+1} \mid w_{<i+1}, h_t) \propto \exp(v_{w_{i+1}} h_{i+1}^t)
\]
where \( v_{w_{i+1}} \) denotes the row of \( V \) corresponding to the word of \( w_{i+1} \). An analogous computation is performed for the previous sentence \( s_{i-1} \).

**Objective.** Given \( (s_{i-1}, s_i, s_{i+1}) \), the objective optimized is the sum of log-probabilities for the next and previous sentences conditioned on the representation of the encoder:
\[
\sum_t \log P(w_{i+1}^t \mid w_{<i+1}^t, h_t) + \sum_t \log P(w_{i-1}^t \mid w_{<i-1}^t, h_t)
\]
The total objective is the above summed over all such training tuples. Adam algorithm \([12]\) is used for optimization.

### 4.2. Visual-semantic embeddings of clips and DVS

The model above describes how to obtain a similarity score between two sentences, whose representations are learned from millions of sentences in books. We now discuss how to obtain similarities between shots and sentences.

Our approach closely follows the image-sentence ranking model proposed by \([13]\). In their model, an LSTM is used for encoding a sentence into a fixed vector. A linear mapping is applied to image features from a convolutional network. A score is computed based on the inner product between the normalized sentence and image vectors. Correct image-sentence pairs are trained to have high score, while incorrect pairs are assigned low scores.

In our case, we learn a visual-semantic embedding between movie clips and their DVS description. DVS ("Descriptive Video Service") is a service that inserts audio descriptions of the movie between the dialogs in order to enable the visually impaired to follow the movie like anyone else. We used the movie description dataset of \([27]\) for learning our embedding. This dataset has 94 movies, and 54,000 described clips. We represent each movie clip as a vector corresponding to mean-pooled features across each frame in the clip. We used the GoogLeNet architecture \([32]\) as well as hybrid-CNN \([38]\) for extracting frame features. For DVS, we pre-processed the descriptions by removing names and replacing these with a *someone* token.

The LSTM architecture in this work is implemented using the following equations. As before, we represent a word embedding at time \( t \) of a sentence as \( x^t \):

\[
i^t = \sigma(W_{xi} x^t + W_{hi} m^{t-1} + W_{ce} e^{t-1})
\]
\[
f^t = \sigma(W_{xf} x^t + W_{hf} m^{t-1} + W_{cf} e^{t-1})
\]
\[
a^t = \tanh(W_{xe} x^t + W_{ae} m^{t-1})
\]
\[
c^t = f^t \circ e^{t-1} + i^t \circ a^t
\]
\[
o^t = \sigma(W_{xo} x^t + W_{ho} m^{t-1} + W_{eo} e^t)
\]
\[
m^t = o^t \circ \tanh(e^t)
\]
where \( \sigma \) denotes the sigmoid activation function and \( (\circ) \) indicates component-wise multiplication. The states \((i^t, f^t, c^t, o^t, m^t)\) correspond to the input, forget, cell, output and memory vectors, respectively. If the sentence is of length \( N \), then the vector \( m^N = m \) is the vector representation of the sentence.

Let \( q \) denote a movie clip vector, and let \( v = W_f q \) be the embedding of the movie clip. We define a scoring function \( s(m, v) = m \cdot v \), where \( m \) and \( v \) are first scaled to have unit norm (making \( s \) equivalent to cosine similarity). We then optimize the following pairwise ranking loss:

\[
\min_\theta \sum_m \sum_k \max \{0, \alpha - s(m, v) + s(m, v_k)\} \\
+ \sum_v \sum_k \max \{0, \alpha - s(v, m) + s(v, m_k)\}
\]
with \( m_k \) a contrastive (non-descriptive) sentence vector for a clip embedding \( v \), and vice-versa with \( v_K \). We train our model with stochastic gradient descent without momentum.

### 4.3. Context aware similarity

We employ the clip-sentence embedding to compute similarities between each shot in the movie and each sentence in the book. For dialogues, we use several similarity measures each capturing a different level of semantic similarity. We compute BLEU [23] between each subtitle and book sentence to identify nearly identical matches. Similarly to [34], we use a tf-idf measure to find near duplicates but weighing down the influence of the less frequent words. 

These similarity measures indicate the alignment between the two modalities. However, at the local, sentence level, alignment can be rather ambiguous. For example, despite being a rather dark book, *Gone Girl* contains 15 occurrences of the sentence “I love you”. We exploit the fact that a match is not completely isolated but that the sentences (or shots) around it are also to some extent similar.

We design a context awareness similarity measure that takes into account all individual similarity measures as well as a fixed context window in both, the movie and book domain, and predicts a new similarity score. We stack a set of \( M \) similarity measures into a tensor \( S(i,j,m) \), where \( i \), \( j \), and \( m \) are the indices of sentences in the subtitle, in the book, and individual similarity measures, respectively. In particular, we use \( M = 9 \) similarities: visual and sentence embedding, BLEU1-5, tf-idf, and a uniform prior. We want to predict a combined score \( \text{score}(i,j) = f(S(I,J,M)) \) at each location \((i,j)\) based on all measurements in a fixed volume defined by \( I \) around \( i \), \( J \) around \( j \), and \( 1, \ldots, M \). Evaluating the function \( f(\cdot) \) at each location \((i,j)\) on a 3-D tensor \( S \) is very similar to applying a convolution using a kernel of appropriate size. This motivates us to formulate the function \( f(\cdot) \) as a deep convolutional neural network (CNN). In this paper, we adopt a 3-layer CNN as illustrated in Figure 3. We adopt the ReLU non-linearity with dropout to regularize our model. We optimize the cross-entropy loss over the training set using Adam algorithm.

### 4.4. Global Movie/Book Alignment

So far, each shot/sentence was matched independently. However, most shots in movies and passages in the books follow a similar timeline. We would like to incorporate this prior into our alignment. In [34], the authors use dynamic time warping by enforcing that the shots in the movie can only match forward in time (to plot synopses in their case). However, the storyline of the movie and book can have crossings in time (Fig. 8), and the alignment might contain giant leaps forwards or backwards. Therefore, we formulate a movie/book alignment problem as inference in a Conditional Random Field that encourages nearby shots/dialog alignments to be consistent. Each node \( y_i \) in our CRF represents an alignment of the shot in the movie with its corresponding subtitle sentence to a sentence in the book. Its state space is thus the set of all sentences in the book. The CRF energy of a configuration \( y \) is formulated as:

\[
- \log p(x,y; \omega) = \sum_{i=1}^{K} \omega_u \phi_u(y_i) + \sum_{i=1}^{K} \sum_{j \in N(i)} \omega_p \psi_p(y_i, y_j)
\]

where \( K \) is the number of nodes (shots), and \( N(i) \) the left and right neighbor of \( y_i \). Here, \( \phi_u(\cdot) \) and \( \psi_p(\cdot) \) are unary and pairwise potentials, respectively, and \( \omega = (\omega_u, \omega_p) \). We directly use the output of the CNN from 4.3 as the unary potential \( \phi_u(\cdot) \). For the pairwise potential, we measure the time span \( d_s(y_i, y_j) \) between two neighbouring sentences in the subtitle and the distance \( d_b(y_i, y_j) \) of their state space in the book. One pairwise potential is defined as:

\[
\psi_p(y_i, y_j) = \frac{(d_s(y_i, y_j) - d_b(y_i, y_j))^2}{(d_s(y_i, y_j) - d_b(y_i, y_j))^2 + \sigma^2}
\]

Here \( \sigma^2 \) is a robustness parameter to avoid punishing giant leaps too harsh. Both \( d_s \) and \( d_b \) are normalized to \([0, 1]\). In addition, we also employ another pairwise potential \( \psi_q(y_i, y_j) = \frac{(d_s(y_i, y_j))^2}{(d_s(y_i, y_j))^2 + \sigma^2} \) to encourage state consistency between nearby nodes. This potential is helpful when there is a long silence (no dialog) in the movie.

**Inference.** Our CRF is a chain, thus exact inference is possible using dynamic programming. We also prune some states that are very far from the uniform alignment (over 1/3 length of the book) to further speed up computation.

**Learning.** Since ground-truth is only available for a sparse set of shots, we regard the states of unobserved nodes as hidden variables and learn the CRF weights with [29].

### 5. Experimental Evaluation

We evaluate our model on our dataset of 11 movie/book pairs. We train the parameters in our model (CNN and CRF)
on *Gone Girl*, and test our performance on the remaining 10 movies. In terms of training speed, our video-text model “watches” 1,440 movies per day and our sentence model reads 870 books per day. We also show various qualitative results demonstrating the power of our approach. We provide more results in the Appendix of the paper.

### 5.1. Movie/Book Alignment

Evaluating the performance of movie/book alignment is an interesting problem on its own. This is because our ground-truth is far from exhaustive – around 200 correspondences were typically found between a movie and its book, and likely a number of them got missed. Thus, evaluating the precision is rather tricky. We thus focus our evaluation on recall, similar to existing work on retrieval. For each shot that has a GT correspondence in book, we check whether our prediction is close to the annotated one. We evaluate recall at the paragraph level, i.e., we say that the GT paragraph was recalled, if our match was at most 3 paragraphs away, and the shot was at most 5 subtitle sentences away. As a noisier measure, we also compute recall and precision at multiple alignment thresholds and report AP (avg. prec.).

The results are presented in Table 4. Columns show different instantiations of our model: we show the leave-one-feature-out setting (0 indicates that all features were used), compare how different depths of the context-aware CNN influence the performance, and compare it to our full model (CRF) in the last column. We get the highest boost by adding more layers to the CNN – recall improves by 14%, and AP doubles. Generally, each feature helps performance. Our sentence embedding (BOOK) helps by 4%, while noisier video-text embedding helps by 2% in recall. CRF which encourages temporal smoothness generally helps (but not for all movies), bringing additional 2%. We also show how a uniform timeline performs on its own. That is, for each shot (measured in seconds) in the movie, we find the sentence at the same location (measured in lines) in the book.

We add another baseline to evaluate the role of context in our model. Instead of using our CNN that considers contextual information, we build a linear SVM to combine different similarity measures in a single node (shot) – the final similarity is used as a unary potential in our CRF alignment model. The Table shows that our CNN contextual model outperforms the SVM baseline by 30% in recall, and doubles the AP. We plot alignment for a few movies in Fig. 8.

**Running Times.** We show the typical running time of each component in our model in Table 5. For each movie/book pair, calculating BLEU score takes most of the time. Note that BLEU does not contribute significantly to the performance and is of optional use. With respect to the rest, extracting visual features VIS (mean pooling GoogleNet features over the shot frames) and SCENE features (mean pooling hybrid-CNN features [38] over the shot frames), takes most of the time (about 80% of the total time).

We also report training times for our contextual model (CNN) and the CRF alignment model. Note that the times are reported for one movie/book pair since we used only one such pair to train all our CNN and CRF parameters. We chose *Gone Girl* for training since it had the best balance between the dialog and visual correspondences.

### 5.2. Describing Movies via the Book

We next show qualitative results of our alignment. In particular, we run our model on each movie/book pair, and visualize the passage in the book that a particular shot in the movie aligns to. We show best matching paragraphs as well as a paragraph before and after. The results are shown in Fig. 8. One can see that our model is able to retrieve a semantically meaningful match despite large dialog deviations from those in the book, and the challenge of matching a visual representation to the verbose text in the book.
Figure 4: Describing movie clips via the book: we align the movie to the book, and show a shot from the movie and its corresponding paragraph (plus one before and after) from the book.

Figure 5: We can use our model to caption movies via a corpus of books. Top: A shot from *American Psycho* is captioned with paragraphs from *Fight Club*, and a shot from *Harry Potter* with paragraphs from *Fight Club*. Middle and Bottom: We match shots from *Avatar* and *Batman Begins* against 300 books from our BookCorpus, and show the best matched paragraph.
5.3. Book “Retrieval”

In this experiment, we compute alignment between a movie and all (test) 10 books, and check whether our model retrieves the correct book. Results are shown in Table 6. Under each book we show the computed similarity. In particular, we use the energy from the CRF, and scale all similarities relative to the highest one (100). Notice that our model retrieves the correct book for each movie.

Describing a movie via other books. We can also caption movies by matching shots to paragraphs in a corpus of books. Here we do not encourage a linear timeline (CRF) since the stories are unrelated, and we only match at the local, shot-paragraph level. We show a description for *American Psycho* borrowed from the book *Fight Club* in Fig. 5.

5.4. The CoCoBook: Writing Stories for CoCo

Our next experiment shows that our model is able to “generate” descriptive stories for (static) images. In particular we used the image-text embedding from [13] and generated a simple caption for an image. We used this caption as a query, and used our sentence embedding trained on books to find top 10 nearest sentences (sampled from a few hundred thousand from BookCorpus). We re-ranked these based on the 1-gram precision of non-stop words. Given the best result, we return the sentence as well as the 2 sentences before and after it in the book. The results are in Fig. 6. Our sentence embedding is able to retrieve semantically meaningful stories to explain the images.

| Mean Recall | AP 1.22 | SVM 0.73 | 0.45 | 0.41 | 0.40 | 0.50 | 0.64 | 0.50 | 0.48 | 1.95 | 5.17 | 17.92 | 19.81 |
|-------------|---------|---------|------|------|------|------|------|------|------|------|------|------|
|              | 2.36    | 10.38  | 12.26 | 12.74 | 11.79 | 11.79 | 12.74 | 11.79 | 11.79 | 14.42 | 28.80 | 5.17 |

Table 4: Performance of our model for the movies in our dataset under different settings and metrics.

<table>
<thead>
<tr>
<th>Mean Recall</th>
<th>AP 3.88</th>
<th>SVM 38.01</th>
<th>52.66</th>
<th>52.95</th>
<th>47.07</th>
<th>48.75</th>
<th>50.03</th>
<th>50.77</th>
<th>52.46</th>
<th>66.77</th>
<th>60.10</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>0.40</td>
<td>10.97</td>
<td>9.62</td>
<td>9.88</td>
<td>5.94</td>
<td>8.57</td>
<td>8.83</td>
<td>9.31</td>
<td>9.64</td>
<td>22.51</td>
<td>23.17</td>
</tr>
</tbody>
</table>

Table 5: Running time for our model per one movie/book pair.

5.5. ConvNets and TF-IDF

<table>
<thead>
<tr>
<th>Mean Recall</th>
<th>AP 3.88</th>
<th>SVM 38.01</th>
<th>52.66</th>
<th>52.95</th>
<th>47.07</th>
<th>48.75</th>
<th>50.03</th>
<th>50.77</th>
<th>52.46</th>
<th>66.77</th>
<th>60.10</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>0.40</td>
<td>10.97</td>
<td>9.62</td>
<td>9.88</td>
<td>5.94</td>
<td>8.57</td>
<td>8.83</td>
<td>9.31</td>
<td>9.64</td>
<td>22.51</td>
<td>23.17</td>
</tr>
</tbody>
</table>

Table 6: Performance of our model for the movies in our dataset under different settings and metrics.

6. Conclusion

In this paper, we explored a new problem of aligning a book to its movie release. We proposed an approach that computes several similarities between shots and dialogues and the sentences in the book. We exploited our new sentence embedding in order to compute similarities between sentences. We further extended the image-text neural embeddings to video, and proposed a context-aware alignment model that takes into account all the available similarity information. We showed results on a new dataset of movie/book alignments as well as several quantitative results that showcase the power and potential of our approach.

Acknowledgments

We acknowledge the support from NSERC, CIFAR, Samsung, Google, and ONR-N00014-14-1-0232. We also thank Lea Jensterle for helping us with elaborate annotation, and Relu Patrascu for his help with numerous infrastructure related problems.

Appendix

In the Appendix we provide more qualitative results.

A. Qualitative Movie-Book Alignment Results

We show a few qualitative examples of alignment in Fig. 8. In this experiment, we show results obtained with our full model (CRF). For a chosen shot (a node in the CRF) we show the corresponding paragraph in the book.
The club was a little emptier than I would have expected for the late afternoon, and the bartender, in red waistcoat and bowtie, was busy wiping down his counter, replacing peanuts and putting out new coasters. A television with the latest La Liga news was hung in an upper corner, and behind him, rows of bottles were reflected in a giant bar mirror. Above the stools, a pergola-type overhead structure held rows of wine glasses. It was a classy place, with ferns in the corner, and not the kind of bar to which I was accustomed. My places usually had a more relaxed feel.

He felt like an idiot for yelling at the child, but his frustration and trepidation was getting the better of him. He glanced toward the shadowed hall and quickly nodded toward Melissa before making his way forward. He came across more children sitting upon a couch in the living room. They watched him, but didn’t move and didn’t speak. His skin started to feel like hundreds of tiny spiders were running up and down it and he hurried on.

A few miles before Tioga Road reached Highway 395 and the town of Lee Vining, Smith turned onto a narrow blacktop road. On either side were parched, grassy open slopes with barbed-wire fences marking property lines. Cattle and horses grazed under trees whose black silhouettes stood stark against the gold-velvet mountains. Marty burst into song: “Home, home on the range, where the deer and the antelope play! Where seldom is heard a discouraging word and the skies are not cloudy all day!”

Figure 6: CoCoBook: We generate a caption for a CoCo image via [13] and retrieve its best matched sentence (+2 before and after) from a large book corpus. One can see a semantic relevance of the retrieved passage to the image.

Figure 7: Alignment results of our model (bottom) compared to ground-truth alignment (top). In ground-truth, blue lines indicate visual matches, and magenta are the dialog matches. Yellow lines indicate predicted alignments.

We can see that some dialogs in the movies closely follow the book and thus help with the alignment. This is particularly important since the visual information is not as strong. Since the text around the dialogs typically describe the scene, the dialogs thus help us ground the visual information contained in the description and the video.

B. Borrowing “Lines” from Other Books

We show a few qualitative examples of top-scoring matches for shot in a movie with a paragraph in another book (a book that does not correspond to this movie).

10 book experiment. In this experiment, we allow a clip in our 10 movie dataset (excluding the training movie) to match to paragraphs in the remaining 9 books (excluding the corresponding book). The results are in Fig. 12. Note that the top-scoring matches chosen from only a small set of books may not be too meaningful.

200 book experiment. We scale the experiment by randomly selecting 200 books from our BookCorpus. The results are in Fig. 15. One can see that by using many more books results in increasingly better “stories”.

Figure 8: **Examples of movie-book alignment.** We use our model to align a movie to a book. Then for a chosen shot (which is a node in our CRF) we show the corresponding paragraph, plus one before and one after, in the book inferred by our model. On the left we show one (central) frame from the shot along with the subtitle sentence(s) that overlap with the shot. Some dialogs in the movie closely follow the book and thus help with the alignment.
The nurse looks at her watch again and pulls a slip of paper out of the folder she's holding, looks at it, and returns it to the folder. She puts the folder down and picks up the log book. Ellis coughs from his place on the wall; she waits until he stops.

"Now. At the close of Friday’s meeting, we were discussing Mr. Harding’s problem. Concerning his young wife. He had stated that his wife was extremely well endowed in the bosom and that this made him uneasy because she drew stares from men on the street." She starts opening to places in the log book; little slips of paper stick out of the top of the book to mark the pages. "According to the notes listed by various patients in the log, Mr. Harding has been heard to say that she 'damn well gives the bastards reason to stare.' He has also been heard to say that he may give her reason to seek further sexual attention. He has been heard to say, 'My dear sweet but illiterate wife thinks any word or gesture that does not smell of brickyard brawn and brutality is a word or gesture of weak dandyism.'"

She continues reading silently from the book for a while, then closes it.

"Certainly, Mr. Cheswick. A vote is now before the group. Will a show of hands be adequate, Mr. McMurphy, or are you going to insist on a secret ballot?""I want to see the hands. I want to see the hands that don’t go up, too."

"Everyone in favor of changing the television time to the afternoon, raise his hand."

I took the envelope and left the rock where Andy had left it, and Andy's friend before him.

Dear Red, If you’re reading this, then you’re out. One way or another, you’re out. And if you've followed along this far, you might be willing to come a little further. I think you remember the name of the town, don’t you? I could use a good man to help me get my project on wheels. Meantime, have a drink on me-and do think it over. I will be keeping an eye out for you. Remember that hope is a good thing, Red, maybe the best of things, and no good thing ever dies. I will be hoping that this letter finds you, and finds you well.

Your friend, Peter Stevens. I didn't read that letter in the field.

Figure 9: **Examples of movie-book alignment.** We use our model to align a movie to a book. Then for a chosen shot (which is a node in our CRF) we show the corresponding paragraph, plus one before and one after, in the book inferred by our model. On the left we show one (central) frame from the shot along with the subtitle sentence(s) that overlap with the shot. Some dialogs in the movie closely follow the book and thus help with the alignment.
"It's very important to us, Mitch," Royce McKnight said warmly.
They all say that, thought McDeere. "Okay, my father was killed in the coal mines when I was seven years old. My mother remarried and lives in Florida. I had two brothers. Rusty was killed in Vietnam. I have a brother named Ray McDeere."

"Where is he?"

Oliver Lambert greeted Mitch and introduced him to the gang. There were about twenty in all, most of the associates in, and most barely older than the guest. The partners were too busy. Lamar had explained, and would meet him later at a private lunch. He stood at the end of the table as Mr. Lambert called for quiet.

"Gentlemen, this is Mitchell McDeere. You’ve all heard about him, and here he is. He is our number one choice this year, our number one draft pick, so to, speak. He is being romanced by the big boys in New York and Chicago and who knows where else, so we have to sell him on our little firm here in Memphis." They smiled and nodded their approval. The guest was embarrassed.

"He will finish at Harvard in two months and will graduate with honors. He’s an associate editor of the Harvard Law Review." This made an impression, Mitch could tell. "He did his undergraduate work at Western Kentucky, where he graduated summa cum laude." This was not quite as impressive. "He also played football for four years, starting as quarterback his junior year." Now they were really impressed. A few appeared to be in awe, as if staring at Joe Namath.

Mitch braced himself and waited.

"Mitch, no lawyer has ever left your law firm alive. Three have tried, and they were killed. Two were about to leave, and they died last summer. Once a lawyer joins Bendini, Lambert & Locke, he never leaves, unless he retires and keep his mouth shut. And by the time they retire, they are a part of the conspiracy and cannot talk. The Firm has an extensive surveillance operation on the fifth floor. Your house and car are bugged. Your phones are tapped. Your desk and office are wired. Virtually every word you utter is heard and recorded on the fifth floor. They follow you, and sometimes your wife. They are here in Washington as we speak. You see, Mitch, The Firm is more than a firm. It is a division of a very large business, a very profitable business. A very illegal business. The Firm is not owned by the partners."

Mitch turned and watched him closely. The Director looked at the frozen pond as he spoke.

Figure 10: **Examples of movie-book alignment.** We use our model to align a movie to a book. Then for a chosen shot (which is a node in our CRF) we show the corresponding paragraph, plus one before and one after, in the book inferred by our model. On the left we show one (central) frame from the shot along with the subtitle sentence(s) that overlap with the shot. Some dialogs in the movie closely follow the book and thus help with the alignment.
Percy slapped the dead man's cheek. The flat smacking sound of his hand made us all jump. Percy looked around at us with a cocky smile on his mouth, eyes glittering. Then he looked back at Bitterbuck again. "Adios, Chief," he said. "Hope hell's hot enough for you."

"Don't do that," Brutal said, his voice hollow and declamatory in the dripping tunnel. "He's paid what he owed. He's square with the house again. You keep your hands off him."

"Aw, blow it out," Percy said, but he stepped back uneasily when Brutal moved toward him. Shadow rising behind him like the shadow of that ape in the story about the Rue Morgue. But instead of grabbing at Percy, Brutal grabbed hold of the gurney and began pushing Arlen Bitterbuck slowly toward the far end of the tunnel, where his last ride was waiting, parked on the soft shoulder of the highway. The gurney's hard rubber wheels moaned on the boards; its shadow rode the bulging brick wall, waxing and waning; Dean and Harry grasped the sheet at the foot and pulled it up over The Chief's face, which had already begun to take on the waxy, characterless cast of all dead faces, the innocent as well as the guilty.

I nodded.

"Oh, yes," Hammersmith said. "He did it. Don't you doubt it, and don't you turn your back on him. You might get away with it once or a hundred times... even a thousand... but in the end..." He raised a hand before my eyes and snapped the fingers together rapidly against the thumb, turning the hand into a biting mouth. "You understand?"

I nodded again.

You said you wouldn't ever leave me.

I know, I'm sorry. You have my whole heart. You always did. You're the best guy. You always were. If I'm not here you can still talk to me. You can talk to me and I'll talk to you. You'll see.

Will I hear you?

You have my whole heart.
Figure 12: Examples of borrowing paragraphs from other books – 10 book experiment. We show a few examples of top-scoring correspondences between a shot in a movie and a paragraph in a book that does not correspond to the movie. Note that by forcing the model to choose from another book, the top-scoring correspondences may still have a relatively low similarity. In this experiment, we did not enforce a global alignment over the full book – we use the similarity output by our contextual CNN.
Figure 13: Examples of borrowing paragraphs from other books – 10 book experiment. We show a few examples of top-scoring correspondences between a shot in a movie and a paragraph in a book that does not correspond to the movie. Note that by forcing the model to choose from another book, the top-scoring correspondences may still have a relatively low similarity. In this experiment, we did not enforce a global alignment over the full book – we use the similarity output by our contextual CNN.
Figure 14: Examples of borrowing paragraphs from other books – 200 book experiment. We show a few examples of top-scoring correspondences between a shot in a movie and a paragraph in a book that does not correspond to the movie. By scaling up the experiment (more books to choose from), our model gets increasingly more relevant “stories”.

---

**Fight.Club**

... ya see, the thing is..." He scratched his beard. "See, I done heard yer little twitter feet up on my ceilin' there, so I come up to do some investigatin'. Yea, that's what I reckon, far as I recall."

Tick exchanged a baffled look with Sofia and Paul. It didn't take a genius to realize they'd already caught Sally in his first lie.

"Well," Tick said, "we need a minute to talk about what we're gonna do."

---

**13th Reality-2**

No, no. I may need to talk to you a little futher, so how about you just let me know if you're gonna leave town.

---

**Fight.Club**

... last night, or were the tears still affecting me more than I realized? I didn’t think about it again. I just turned and walked to the bathroom. A quick shower and we’d be on our way to the airport.

Twenty minutes later I was ready, my hair still soaking wet. I was dressed in a pair of navy blue dress slacks, an emerald green silk blouse, and a navy suit jacket that matched the pants. Jeremy had also chosen a pair of black low-heeled pumps and included a pair of black thigh-highs. Since I didn’t own any other kind of hose, that I didn’t mind. But the rest of it...

"Next time you pick out clothes for me to run for my life in, include some jogging shoes. Pumps, no matter how low-heeled, just aren’t made for it."

---

**AKissofShadows**

Two pair of black pants? - Yes, sir.

---

**The.Green.Mile**

You, he wanted to say, I'm thinking of you. I'm thinking of your stink and how bad you smell and how I can't stop smelling you. I'm thinking of how you keep staring at me and how I never say anything about it and I don't know why. I'm thinking of you staring at me and why someone's screaming at me inside my head and how someone's screaming inside my head and why it seems odd that I'm not worried about that.

---

**Aeons-Gate-1**

I'm thinking I don't know what I would do if you were gone.
Figure 15: Examples of borrowing paragraphs from other books – 200 book experiment. We show a few examples of top-scoring correspondences between a shot in a movie and a paragraph in a book that does not correspond to the movie. By scaling up the experiment (more books to choose from), our model gets increasingly more relevant “stories”. Bottom row: failed example.
C. The CoCoBook

We show more results for captioning CoCo images [18] with passages from the books.

“somewhere you ’ll never find it , ” owens sneered . if never meant five seconds , his claim was true . the little shit ’s gaze cut left , where a laptop sat on a coffee table . trey strode to it . owens ’ email program was open .

seriously . its like a train crashing into another train . a train wreck . just something like that . i try to convince her .

everyone was allowed to rest for the next twenty-four hours . that following evening : the elect , not their entourages , were called to a dining hall for supper with lady dolorous . a table that curved inward was laden with food and drink . the wall behind the table was windows with a view of the planet . girls in pink stood about and at attention .

he had simply ... healed . brian watched his fellow passengers come aboard . a young woman with blonde hair was walking with a little girl in dark glasses . the little girl ’s hand was on the blonde ’s elbow , the woman murmured to her charge , the girl looked immediately toward the sound of her voice , and brian understood she was blind - it was something in the gesture of the head .
this was a beautiful miniature reproduction of a real London town house, and when Jessamine touched it, Tessa saw that the front of it swung open on tiny hinges. Tessa caught her breath. There were beautiful tiny rooms perfectly decorated with miniature furniture, everything built to scale, from the little wooden chairs with needlepoint cushions to the cast-iron stove in the kitchen. There were small dolls, too, with china heads, and real little oil paintings on the walls. “This was my house.”

If he had been nearby he would have dragged her out of the room by her hair and strangled her. During lunch break she went with a group back to the encampment. Out of view of the house, under a stand of towering trees, several tents were sitting in a field of mud. The rain the night before had washed the world, but here it had made a mess of things. A few women fired up a camp stove and put on rice and lentils.

Then a frightened yell. “Hang on!” Suddenly, Jake was flying through the air. Nefertiti became airborne, too. He screamed, not knowing what was happening—then he splashed into a pool of water.

Grabbing his wristwatch off the bedside table he checked the time, grimacing when he saw that it was just after two in the afternoon. Jeanne Louise should n’t be up yet. Stifling a yawn, he slid out of bed and made his way to the en suite bathroom for a shower. Twenty minutes later Paul was showered, dressed, and had brushed his teeth and hair. Feeling somewhat alive now, he made his way out of his and Jeanne Louise’s room, pausing to look in on Livy as he passed.
she cried. Quentin put a heavy, warm, calming hand on her thigh, saying, “he should be sober by then.” A cell phone rang. He pulled his from his back pocket, glanced at it, then used the remote to turn the TV to the channel that showed the feed from the camera at the security gate. “Oh, it’s Rachel.”

Now however she was out of his shot. He had missed it completely until he had ended up on the ground with his shotgun. An old clock hung on the wall near the door. The was obviously broken, the small red hand ticking the same second away over and over again. Morgan squeezed the trigger and pellets ripped out of their package, bounced down the barrel, flew through the air and ripped into the old clock tearing it in two before it smashed to the ground.

A man sat in a chair, facing the wall opposite of me. It nearly startled me when I first saw him, and made a bit of a squeak, but he did nothing. He had dark gray hair, a black suit and pants, and a gray and blue striped tie. S-sir? I said.

It’s been years since we last played together, but as I recall, he was rather weak at the net. Or was it his serving? All I know is he plays tennis much better than he plays cricket. Perhaps, Mr Brearly, Frances eventually replied, we should wait until we actually start playing. Then we can ascertain our opposition’s faults, and make a plan based on the new information.
since it was the middle of summer, there were candles in the
fireplace instead of a fire. But it still cast a romantic glow over
the room. There were candles on the mantle and on a table set
up in the corner with flowers. As she looked around, her eyes
instinctively turned to find Max who was behind a bar opening a
bottle of champagne. The doors were closed quietly behind her
and her mouth felt dry as she looked across the room at the man
who had haunted her dreams for so long.

The open doorway of another house provided a view of an ancient
game of tiles. It wasn't the game that held Redding's attention.
It was the four elderly people who sat around a table playing
the game. They were well beyond their productive years and the
canal township had probably been their whole lives. Redding and
Lin Ming stepped away from the doorway right into the path of a
wooden pushcart.

Along with the fish, Howard had given them some other picnic
treats that had spoiled... mushrooms in cream sauce, rotted
greens. The bats and Tem were only eating from the river now,
but the remaining picnic food was running low. There were a few
loaves of stale bread, some cheese, some dried vegetables, and
a couple of cakes. Gregor looked over the supplies and thought
about boots wailing for food and water in the jungle. It had been
unbearable.

He felt the first stirrings of fear mixing with his anger. A light
flicked on in the room and Eric jerked, blinking for a minute
at the brightness before the images focused. There was a tall,
thin man standing over a mannequin. He looked like he was
assembling it, since its leg was on the ground next to the man
and its arm was in two pieces farther away. Then the mannequin
's head turned.
References