Abstract

Automatic story comprehension is a fundamental challenge in Natural Language Understanding, and can enable computers to learn about social norms, human behavior and commonsense. In this paper, we present a story comprehension model that explores three distinct semantic aspects: (i) the sequence of events described in the story, (ii) its emotional trajectory, and (iii) its plot consistency. We judge the model’s understanding of real-world stories by inquiring if, like humans, it can develop an expectation of what will happen next in a given story. Specifically, we use it to predict the correct ending of a given short story from possible alternatives. The model uses a hidden variable to weigh the semantic aspects in the context of the story. Our experiments demonstrate the potential of our approach to characterize these semantic aspects, and the strength of the hidden variable based approach. The model outperforms the state-of-the-art approaches and achieves best results on a publicly available dataset.

1 Introduction

Narratives are a fundamental part of human language and culture. They serve as vehicles to share experiences, information and goals. For these reasons, automatically understanding stories is an interesting but challenging task for Computational Linguists (Mani, 2012). Story comprehension involves not only an array of NLP capabilities, but also some common sense knowledge and an understanding of normative social behavior (Charniak, 1972). Past research has focused on various aspects of story understanding such as identifying character personas (Bamman et al., 2014; Valls-Vargas et al., 2015), interpersonal relationships (Chaturvedi, 2016), plot patterns (Jockers, 2013), narrative structures (Finlayson, 2012). There has also been an interest in predicting what is expected to happen next in a piece of text (Chambers and Jurafsky, 2008). Human readers are good at filling-in-the-gaps or inferring information that is not explicitly stated in the text. However, computers are not yet able to match their performance on predicting what could be the likely next step in a given sequence of events described in a story.

Recently, Mostafazadeh et al. (2016) introduced the story-cloze task for testing this ability, albeit without the aspect of language generation. This task requires choosing the correct ending to a given four sentences long story (also referred to as context) from two provided alternatives. Fig. 1 shows an example story consisting of a short context, and two ending options.

In this work we address this story-cloze task. While the short nature and third person narrative style of these stories help us circumvent the problem of speaker identification and processing long
dialogues, the crowdsourced dataset ensures that they reflect real-world and commonsense stories. Our approach emphasizes the joint contribution of multiple aspects to story understanding, which future research can build upon.

In this paper we explore three semantic aspects of story understanding: (i) the sequence of events described in the story, (ii) the evolution of sentiment and emotional trajectories, and (iii) topical consistency. The first aspect is motivated from approaches in semantic script induction, and evaluates if events described in an ending-alternative are likely to occur within the sequence of events described in the preceding context. For example, in the story in Fig. 1, Wesley gets angry and bites his sister’s hand. So, a next likely step might suggest that he would be scolded. However, there are multiple semantic aspects to story understanding beyond analyzing events and scripts. Stories often describe characters (e.g. Wesley) who need to be viewed as social and emotional agents. They not only describe events involving these characters, but also reflect their social lives and emotional states. Our model captures this by evaluating if the sentiment described in an ending option makes sense considering the context of the story. For example, in the story in Fig. 1, the general sentiment of being scolded is better aligned with the sentiment of Wesley being angry and jealous, compared to that of being nice. Also, stories generally revolve around coherent themes and topics. Our model accounts for that by analyzing if the topic of an ending option is consistent with the preceding context. We present a log-linear model that is used to weigh the various aspects of the story using a hidden variable. It then uses this hidden variable to predict the correct ending for the given story.

We demonstrate the strength of our approach by comparing it with the existing state-of-the-art methods for this task. We first validate the predictive potential of the features that correspond to the three semantic aspects through a simple classifier trained using these features. We then demonstrate the benefit of using our hidden variable approach by showing that it significantly outperforms the above mentioned classifier and other baselines, and achieves an accuracy of 77.60% on the task. Our key contributions are:

- We model story understanding as a joint model over multiple semantic aspects, and utilize the idea for predicting a story’s end.
- We design linguistic features that incorporate world knowledge and narrative awareness.
- We present a hidden variable approach to weigh these aspects in a story’s context.
- We empirically demonstrate that our approach significantly outperforms state-of-the-art methods.

2 Predicting Story Ending

Given an $L$ sentences long context, $c = (c_1, c_2, c_3 \ldots c_L)$, and two ending-options, $o_1$ and $o_2$, we aim to predict which ending option forms an inconsistent story. This is a binary classification task. We assume that the inconsistency can arise from one (or more) of certain semantic aspects. In this section, we first describe the intuition behind using these aspects and the features that we designed to capture them (Sec. 2.1). We then describe our model which uses a latent variable to weigh these aspects in light of the story, and then predicts its ending (Sec. 2.2).

2.1 Measuring Consistency

Our approach analyzes the following aspects of story understanding: Event-sequence, Sentiment-trajectory, and Topical Consistency.

Event-sequence: For a story, or any piece of text, to be coherent, it needs to describe a meaningful or ‘mutually entailing’ sequence of events (Chatman, 1980). For instance, in Figure 1 Wesley got angry → Wesley bit her hand → Wesley was scolded describes a more coherent sequence of events, as compared to Wesley got angry → Wesley bit her hand → Wesley got a cookie

Prior work in script-learning attempts to model such prototypical sequence of events (usually captured through verbs). For this task, we wanted to model events at an abstraction level that would be generalizable and yet semantically meaningful. Peng and Roth (2016) recently proposed a neural SemLM approach, to model such sequence of events using a language model of FrameNet (Baker et al., 1998) frames that are evoked in the given text. It represents an event using the corresponding predicate frame and its sense, obtained using a Semantic Role Labeler (Punyakanok et al., 2004). It also extends the frame definition to include explicit discourse markers (such as but, and) since they model relationships between frames. For example, in Fig-
ure 1, the SemLM representation for the last sentence of the context is ‘Get.01-and-bit.01’. Here, ‘01’ indicates specific predicate senses for verbs ‘get’ and ‘bit’ with ‘and’ being a discourse marker. Also, it produces ‘scold.01’ and ‘give.01’ for the correct and incorrect endings respectively. We train this language model using a log bilinear language model (Mnih and Hinton, 2007) on a collection of unannotated short stories (see Sec. 3.1) and also 20 years of New York Times data\(^1\).

Given a sequence of frames evoked in the context, such a trained language model can then be used to get the conditional probabilities of the frame(s) evoked in each of the two ending-options. The option with more probable frame(s) is likely to be the appropriate ending. With this intuition in mind, for each of the two ending-option, \(o_i\), we design features whose values are probabilities of frames evoked in that option \((f_{o_i})\), given the sequence of frames, \(\langle f_1, f_2, \ldots, f_D \rangle\), evoked in the context, \(c = \langle c_1, c_2, c_3 \ldots c_L \rangle\). We consider increasingly longer frame-contexts for conditional probability computation, i.e. for each option, \(o_i\), we extract the following features: \(P(f_{o_i}|f_D)\), \(P(f_{o_i}|f_Df_{D-1})\), \(\ldots\), \(P(f_{o_i}|f_Df_{D-1} \ldots f_1)\). For each of these features, we additionally also include a comparative binary feature whose value is 1 if the conditional probability of one of the options \(o_2\) is greater than the corresponding conditional probability of the other option \(o_1\) (E.g. \(P(o_2|f_D) > P(o_1|f_D)\)), and −1 otherwise. Our preliminary experiments indicated that these features were helpful for supervised classification.

**Sentiment-trajectory:** As mentioned before, stories are different from objective texts such as news articles, as they additionally describe sentiments or emotions. Some stories can be categorized as happy stories while others as sad. However, most stories depict evolving sentiments in their plots as they progress (Vonnegut, 1981).

With the goal of modeling such sentiment trajectories, we assumed that a story can be divided into the following narrative-segments: a *beginning*, a *body*, a *climax*, and an *ending*. While this narrative-segmentation process warrants deeper research, in this paper we adopt a simple methodology. We treat the first sentence of the \(L\) sentences long context as the *beginning*, the next \(L - 2\) sentences are treated as the *body*, the last sentence of the context forms the *climax*, and the two options form the (possible) *ending*.\(^2\) We then assigned a positive, negative, or neutral sentiment to each segment, represented as \(S\text{(segment)} = \text{sign}(\text{number of positive words} - \text{number of negative words})\) in the segment. The sentiment polarity of a word was determined by a look-up from pre-trained sentiment lexica (Liu et al., 2005; Wilson et al., 2005)\(^3\). Thus, the \(L\) length context can now be viewed as a sequence of its segment’s sentiments. Lastly, we learn sentiment trajectories in form of N-gram language models from an unannotated corpus of short stories (Sec. 3.1) that learn: (i) \(P(S\text{(ending)}|S\text{(climax)}, S\text{(body)}, S\text{(beginning)})\); (ii) \(P(S\text{(ending)}|S\text{(climax)}, S\text{(body)})\); and (iii) \(P(S\text{(ending)}|S\text{(climax)})\).

The process described above learns typical sentiment trajectories over narrative-segments. However, it does not model a story’s overall sentiment (i.e. whether it is a happy or a sad story, in general). To capture this notion, we train another language model to learn \(P(S\text{(ending)}|S\text{(context)})\), where \(S\text{(context)}\) is the sentiment of the full context (without segmentation).

Finally, for each ending option, we extract features whose values are the four conditional probabilities described above. As before we also consider four comparative binary features.

**Topical Consistency:** This aspect is motivated by the idea that stories are topically cohesive (Bamberg, 2012), and in a typical story, new topics (concepts, entities or ideas) are not introduced towards the end because it does not allow the story-writer enough narrative space and time to develop and describe them (Jovchelovitch and Bauer, 2000). We capture the notion of topic of a sentence using *topic-words* (the nouns and verbs appearing in it (Lapata and Barzilay, 2005)). For each option, we first *align* each of its topic-words with the most similar topic-word in one of the context-sentences, while defining the alignment score as this similarity value. We measure similarity between two words using the cosine similarity of their vector space representations (using

\(^1\)Owing to the large size of the training data and the fact that we abstract to the frame-semantic (and not verb) level, we cover most instances (76%) in our dataset.

\(^2\)The reported segmentation process made sense from qualitative analysis on a random sample, and also led to superior performance compared to alternate strategies.

\(^3\)Polarities of ‘negated’ word were reversed (determined from *neg* dependency relation in the corresponding sentence).
pretrained GloVe (Pennington et al., 2014) vectors). We then quantify the topical-closeness of an ending option with the context using averaged alignment score of its topic-words\(^4\). For each ending option, we extract one feature whose value is this topical-closeness with the context. As before, we also include a binary comparative feature.

2.2 Hidden Coherence Model

Sec. 2.1 described the three semantic consistency aspects and the corresponding features. We now describe our model which uses these features (represented as \(\vec{f}_{co}\) in the rest of this paper) to identify the (in)coherent ending-option. The model is also dependent on another feature set, \(\vec{\phi}_{co}\), which will be discussed later in this section.

Formally, our model addresses the following binary classification problem: given the multisetence context, \(c\), and two ending-options, \(o_1\) and \(o_2\), predict the answer, \(a \in \{0, 1\}\). The correct ending for the story is \(o_1\) when \(a = 0\) and \(o_2\) otherwise. Our training data consists of instances (context and ending options) labeled with corresponding answers \(a\). It does not contain any other annotation (like semantic consistency aspects).

The model proceeds by assuming that there are \(K\) different semantic consistency aspects and that an ending-option can lead to an incoherent story by violating any of these aspects (our implementation uses \(K = 3\) corresponding to the three aspects described in Sec. 2.1). The model achieves this by assuming that each instance belongs to a latent category, \(z \in \{1, 2, 3 \ldots K\}\), which advises the model on the importance of these aspects for the given instance. Using these definitions and assumptions, the probability of an answer given the context and the ending-options can be modeled as:

\[
P(a|c, o_1, o_2) = \sum_{z} P(z|c, o_1, o_2) P(a|z, c, o_1, o_2)
\]

We parameterize \(P(z|c, o_1, o_2)\) as:

\[
P(z|c, o_1, o_2) = \frac{e^{-\vec{\lambda}_z \vec{\phi}_{co}}}{\sum_{k} e^{-\vec{\lambda}_k \vec{\phi}_{co}}}
\]

where, \(\vec{\phi}_{co}\) is the feature vector used for assigning a value to the hidden variable for an instance, and \(\vec{\lambda}_z\) is the weight vector of the log-linear model for the \(z^{th}\) aspect. There are \(K\) weight vectors, one corresponding to each of the \(K\) aspects.

For predicting the answer, \(a\), we assume that each aspect has a separate logistic-regression based prediction model parameterized as:

\[
P(a|z, c, o_1, o_2) = \frac{(e^{-\vec{\lambda}_z \vec{f}_{co}})^{1-a}}{1 + e^{-\vec{\lambda}_z \vec{f}_{co}}}
\]

where \(\vec{f}_{co}\) is the feature vector constructed from the context and ending-options for the \(z^{th}\) aspect, and \(\vec{\lambda}_z\) are the corresponding weights.

**Training:** The model parameters, \(\vec{\lambda}_z\) and \(\vec{\lambda}_x\), are learned during the training process by maximizing the log-likelihood of the data. We use Expectation-Maximization (Dempster et al., 1977) for training. During the E-step we compute the expectations for latent variable assignments using parameter values from the previous iteration as:

\[
< z_n^k > \propto \frac{e^{-\vec{\lambda}_z \vec{\phi}_{co}} P(a_n|z_n^k, c_n, o_{1n}, o_{2n})}{\sum_{k'} e^{-\vec{\lambda}_{k'} \vec{\phi}_{co}}}
\]

where, a subscript of \(n\) represents the \(n^{th}\) training instance out of a total of \(N\) instances. \(z_n^k\) represents \(n^{th}\) instance getting assigned to the \(k^{th}\) aspect, and \(< >\) denotes expected values.

In the M-step, given the expected assignments, we maximize the following expected log complete likelihood with respect to the model parameters using gradient ascent:

\[
< L > = \sum_n \sum_k < z_n^k > \left( \log \frac{e^{-\vec{\lambda}_z \vec{\phi}_{co}}}{\sum_{k'} e^{-\vec{\lambda}_{k'} \vec{\phi}_{co}}} + \log \frac{(e^{-\vec{\lambda}_z \vec{f}_{co}})^{1-a_n}}{1 + e^{-\vec{\lambda}_z \vec{f}_{co}}} \right)
\]

**Features:** Our model uses two types of features:

(i) for aspect-specific prediction model, \(\vec{f}_{co}\), and (ii) for hidden aspect assignment, \(\vec{\phi}_{co}\). The features extracted for each of the \(K = 3\) aspects, \(\vec{f}_{co}\), were described in Sec. 2.1. For the hidden aspect assignment, we needed features that could analyze the two options in light of the given context, and characterize the importance of various aspects for the given instance. One way to measure an aspect’s importance is by quantifying how different the two options are with respect to that aspect. The underlying assumption is that the option that
summarize, for each instance, we define the difference between this value for the two options. To measure its topical-closeness, we project the context and the ending-options into the same vector space. Based on these vector representations, it predicts the ending-option with the largest cosine similarity with the context.

Msap: The task addressed in this paper was also a shared task for an EACL’17 workshop and this baseline (Schwartz et al., 2017) represents the best performance reported on its leaderboard (Mostafazadeh et al., 2017). It trains a logistic regression based on stylistic and language-model based features.

LR: Our next baseline is a simple logistic regression model which is agnostic to the fact that there are multiple types of aspects. Given a context and ending-options, it predicts the answer using the same features (Sec. 2.1) as the Hidden Coherence model but clubs them all into one feature-vector.

Majority Vote: This ensemble method uses the features extracted for each of the $K = 3$ aspects, to train $K$ separate logistic regression models. It then makes a prediction by taking a majority vote of these $K$ classifiers.

Soft Voting: This baseline also learns $K$ different aspect-specific classifiers. However, instead of taking a majority vote, it computes a score for each option, $o_i$, as $\prod_{k}^K P_k(\text{ending} = o_i|e, o_1, o_2)$. Here $P_k$ represents the probability obtained from the $k^{th}$ logistic regression. The final prediction corresponds to the option with greater score.

Aspect-aware Ensemble: Like the voting methods, this baseline also trains $K$ different aspect-specific classifiers. However, it makes the final prediction by training another logistic regression over their predictions.

3.3 Quantitative Results

Table 1 shows accuracies of various models on the held-out test set. An always-one classifier would get 51.3% accuracy on the task and human performance is reported to be 100% (Mostafazadeh et al., 2016).
Table 1: Test-set accuracies of various models.

<table>
<thead>
<tr>
<th>Model</th>
<th>Accuracy</th>
</tr>
</thead>
<tbody>
<tr>
<td>DSSM (Mostafazadeh et al., 2016)</td>
<td>58.5%</td>
</tr>
<tr>
<td>Msap (Schwartz et al., 2017)</td>
<td>75.2%</td>
</tr>
<tr>
<td>Majority Voting</td>
<td>69.5% *</td>
</tr>
<tr>
<td>Aspect aware ensemble</td>
<td>71.5% *</td>
</tr>
<tr>
<td>LR</td>
<td>74.4% *</td>
</tr>
<tr>
<td>Soft Voting</td>
<td>75.1%</td>
</tr>
<tr>
<td>Hidden Coherence Model</td>
<td>77.6% *</td>
</tr>
</tbody>
</table>

Table 2: Performance comparison of various aspect features. Our event-sequence based features are most helpful followed by Sentiment-trajectory and then Topical Consistency based features.

<table>
<thead>
<tr>
<th>Features</th>
<th>Accuracy</th>
</tr>
</thead>
<tbody>
<tr>
<td>All</td>
<td>74.4%</td>
</tr>
<tr>
<td>Event-sequence</td>
<td>71.6%</td>
</tr>
<tr>
<td>Sentiment</td>
<td>64.5%</td>
</tr>
<tr>
<td>Topic</td>
<td>55.2%</td>
</tr>
</tbody>
</table>

et al., 2016). A * indicates that the model’s accuracy was significantly better than the previous best model in the table (using McNemar’s test with \( \alpha = 0.1 \)). We can see that the logistic regression, LR, outperforms the DSSM model indicating the strength of the features extracted for the various story-consistency aspects. Also, the Soft Voting approach gives us slight benefit over the LR model, possibly because of increased expressivity which includes better organization of features into groups or aspects. Majority Vote, in spite of sharing a similar classifier structure, does not perform as well. This might happen because it takes a hard vote of individual classifiers, which might be detrimental to model performance if one of the classifiers is weak. Our analysis in Sec. 3.4 shows that the topical-consistency features indeed result in a relatively weak classifier. The Aspect aware Ensemble performs better possibly because of its ability to weight the aspects (though not in context of the story).

Lastly, we can see that the proposed Hidden Coherence model, with an accuracy of 77.60%, outperforms all other models. The superior performance of our model indicates the benefit of the context-sensitive weighing of individual consistency aspects.

3.4 Ablation Study

We now investigate the predictive value of the various aspect-specific features. Table 2 shows the performance of a logistic regression model trained using all the features (All) and then using individual feature-groups. We can see that the features extracted from the aspect analyzing the event-sequence have the strongest predictive power, followed by those characterizing Sentiment-trajectory. The features measuring topical consistency result in lowest accuracy but they still perform better than random on the task.

3.5 Qualitative Results

Table 3 shows example stories, and weights given to the three aspects. An aspect’s weight is its contribution towards the predicted output, and is shown as a bar of vertically stacked blocks in the last column. A block’s height is proportional to its aspect’s weight. Light grey block represents Event-sequence, and dark grey and black blocks represent Sentiment-trajectory and Topical consistency respectively.

The first row describes the story of a man hurting himself. A human reader can guess from commonsense knowledge that people usually recover (correct ending) after being hurt and do not repeat their mistake (incorrect ending). Accordingly, our model also primarily used the aspect analyzing events in this story, which is indicated by the long light grey block in its weight bar. Also, we can see that the topic of both the options is consistent with the story, and the model gave a very small weight to the Topical Consistency aspect indicated by the almost indiscernible black block in its weight bar. Similarly, the second row describes the story of Pam being proud of her yard work. There is a striking sentimental contrast between the two options (upset versus satisfied), and the model relies primarily on sentiments (dark grey). The last row, describes the story of Maria making candy apples. The incorrect ending introduces a new entity/idea, apple pie, resulting in topical incoherence of this option with the rest of the story. The model relies primarily on topic (black) and events (light grey). Reliance on events makes sense because it is likely for a person to enjoy what they fondly cook. The model gave a weight of 40% to the topical aspect, which is high as compared to its average weight across the dataset.
He didn’t know how the television worked. He tried to fix it, anyway. He climbed up on the roof and fiddled with the antenna. His foot slipped on the wet shingles and he went tumbling down.

Pam thought her front yard looked boring. So she decided to buy several plants. And she placed them in her front yard. She was proud of her work.

Maria smelled the fresh Autumn air and decided to celebrate. She wanted to make candy apples. She picked up the ingredients at a local market and headed home. She cooked the candy and prepared the apples.

<table>
<thead>
<tr>
<th>Context</th>
<th>Incorrect Ending</th>
<th>Correct Ending</th>
<th>Weights</th>
</tr>
</thead>
<tbody>
<tr>
<td>He didn’t know how the television worked. He tried to fix it, anyway. He climbed up on the roof and fiddled with the antenna. His foot slipped on the wet shingles and he went tumbling down.</td>
<td>He decided that was fun and to try tumbling again.</td>
<td>Thankfully, he recovered.</td>
<td></td>
</tr>
<tr>
<td>Pam thought her front yard looked boring. So she decided to buy several plants. And she placed them in her front yard. She was proud of her work.</td>
<td>Pam was upset at herself.</td>
<td>Pam was satisfied.</td>
<td></td>
</tr>
<tr>
<td>Maria smelled the fresh Autumn air and decided to celebrate. She wanted to make candy apples. She picked up the ingredients at a local market and headed home. She cooked the candy and prepared the apples.</td>
<td>Maria’s apple pie was delicious.</td>
<td>She enjoyed the candy apples.</td>
<td></td>
</tr>
</tbody>
</table>

Table 3: Examples of stories, ending-options, and aspect weights learned by our model. Aspect weights are shown as bars of stacked blocks in the last column (light grey, dark grey and black represent Event-sequence, Sentiment-trajectory and Topical Consistency respectively). A block’s height is proportional to its component’s weight. Black blocks are sometimes not visible because there were too small.

3.6 Discussion

Error Analysis: Table 4 shows examples of stories for which our model could not predict the correct ending. We believe that many of these stories require a deeper understanding of language and commonsense. For example, in the story described in the first row, the protagonist accepted an invitation from his friends to go to a club but danced terribly, and so he was asked to stay home the next time. To make the correct prediction in this story, the model not only needs to understand that if one does not dance well at a club they are likely to be not invited in the future, but also that staying home is the same as not getting invited. Similarly, the second row shows a story in which Johnny asks Anita out, but she makes an excuse. He later sees her with another guy and decides not to ask her out again. This example requires identifying that Anita’s excuse was a lie indicating her disinterest in Johnny, which makes it unlikely for Johnny to invite her again. It also needs an understanding of inter-personal relationships, i.e. seeing a potential lover with another person leads to estrangement.

Social Analysis: To further explore the significance of social relations in stories, we consider the special case of romantic stories. We use a deterministic heuristic to identify romantic stories using lexical matches with a handcrafted list containing words like marry, proposal, girlfriend, ask out, etc. We then applied the following two rules: (i) if a story contains two characters, then output the option whose sentiment matches that of the context, (ii) if a story contains three characters, then output the option with negative sentiment. Most stories in our dataset contained few characters. These rules are motivated by the intuition that a romantic story between two people can have a happy or sad ending depending on the context. However, a romantic story with three people is likely to describe a love triangle, and so not end well. Expectedly, these rules had low coverage (of about 60 stories), but a considerably high accuracy (70%) when active. Furthermore, a closer analysis revealed that most errors resulted from incorrect coreference resolutions (leading to incorrect count of characters). This indicates the utility of understanding semantics of social relationships for story comprehension and it could potentially be another aspect to consider while solving such tasks.

Sentiment Analysis: We now explore the insights obtained by modeling sentiments in stories. Mostafazadeh et al. (2016) presented two baselines for this task whose outputs were simply the ending whose sentiment agreed with (i) the complete story, or (ii) the climax (last sentence of the story). While their performances were close to random, our sentiment based features yield a much higher accuracy of 64.5% (see Table 2). This could possibly be attributed to our approach’s ability to learn such rules from the data itself, rather than making hard assumptions. For instance, our language model of overall narrative sentiments in-
My friends all love to go to the club to dance. They think it’s a lot of fun and always invite. I finally decided to tag along last Saturday. I danced terribly and broke a friend’s toe.

The next weekend, I was asked to please stay home. My friends decided to keep inviting me as I am so much fun.

Johnny thought Anita was the girl for him, but he was wrong. He invited her out but she said she didn’t feel well. Johnny decided to go to a club, just to drink and listen to music. At midnight, he looked back and saw Anita dancing with another guy.

Johnny did not ask Anita out again. Johnny wanted to ask Anita out again.

<table>
<thead>
<tr>
<th>Context</th>
<th>Incorrect Ending</th>
<th>Correct Ending</th>
</tr>
</thead>
<tbody>
<tr>
<td>My friends all love to go to the club to dance. They think it’s a lot of fun and always invite. I finally decided to tag along last Saturday. I danced terribly and broke a friend’s toe.</td>
<td>The next weekend, I was asked to please stay home.</td>
<td>My friends decided to keep inviting me as I am so much fun.</td>
</tr>
<tr>
<td>Johnny thought Anita was the girl for him, but he was wrong. He invited her out but she said she didn’t feel well. Johnny decided to go to a club, just to drink and listen to music. At midnight, he looked back and saw Anita dancing with another guy.</td>
<td>Johnny did not ask Anita out again.</td>
<td>Johnny wanted to ask Anita out again.</td>
</tr>
</tbody>
</table>

Table 4: Examples of stories incorrectly predicted by our model.

dicates that while happy stories mostly have happy endings (with a conditional probability of 74%), the reverse is not true. In particular, sad stories (with overall negative sentiments) end with a negative sentiment in only 52% of the cases. We made similar observations regarding sentimental conformity between endings and climaxes.

Our features’ superior performance can also be attributed to their deeper understanding of not just overall sentiments but also their trajectories. Our language models indicate that stories that exhibit a positive sentiment in all three narrative segments (beginning, body, and climax) have very high chance of happy endings (∼83%). Similarly, stories with negative sentiments in the three segments also have a fair chance of having sad endings (60%). This is different from stories with an overall negative sentiment, in which case the sentiment may be exhibited in only certain narrative segments. The language models also identify a pattern of hopeful stories, in which the sentiment begins as negative but moves towards positive in the body and climax, resulting in mostly happy endings (∼70%). This was not true for the reverse case: pessimistic stories with positive beginning but negative body (and/or climax) were equally likely to have positive or negative endings (52%). Supplementary material contains sample stories for each of the above observations.

4 Related Work

We now review previous work done in this field. Our work touches upon several research areas.

4.1 Story understanding:

Our work is most closely related to the field of narrative understanding. Apart from event-centric understanding of narrative plots (Lehnert, 1981; McIntyre and Lapata, 2010; Goyal et al., 2010; Elsner, 2012; Finlayson, 2012), recent methods have focused on understanding narratives from the perspective of characters (Wilensky, 1978) mentioned in them. These methods study character personas (Bamman et al., 2013, 2014) or Proppian (Propp, 1968) roles (Valls-Vargas et al., 2014, 2015), inter-character relationships (Iyyer et al., 2016; Chaturvedi et al., 2016, 2017), and social networks of characters (Elson et al., 2010; Elson, 2012; Agarwal et al., 2013, 2014; Krishnan and Eisenstein, 2015; Srivastava et al., 2016).

4.2 Events-centered learning:

Our Entity-sequence component is closely related to semantic script learning. Script learning focuses on representing text using a prototypical sequences of events, their participants and causal relationships between them, called scripts (Schank and Abelson, 1977; Mooney and DeJong, 1985). Several statistical methods have been proposed to automatically learn scripts or scripts-like structures from unstructured text (Chambers and Jurafsky, 2008, 2009; Jans et al., 2012; Orr et al., 2014; Pichotta and Mooney, 2014). Such methods for script-learning also include Bayesian approaches (Bejan, 2008; Fermann et al., 2014), sequence alignment algorithms (Regneri et al., 2010) and neural networks (Modi and Titov, 2014; Granroth-Wilding and Clark, 2016; Pichotta and Mooney, 2016). There has also been work on representing events in a structured manner using schemas, which are learned probabilistically (Chambers, 2013; Cheung et al., 2013; Nguyen et al., 2015), using graphs (Balasubramanian et al., 2013) or neural approaches (Titov and Khoddam, 2015). Recently, Ferraro and Durme (2016) presented a unified Bayesian model for
scripts and frames.

4.3 Textual Coherence:

Our work is also related to the study of coherence in discourse. A significant amount of prior work is primarily based on the Centering Theory Framework (Grosz et al., 1995) and focus on entities and their syntactic roles (Karamanis, 2003; Karamanis et al., 2004; Lapata and Barzilay, 2005; Barzilay and Lapata, 2008; Elsner and Charniak, 2008). Other approaches measure coherence using topic drift within a domain (Barzilay and Lee, 2004; Fung and Ngai, 2006), co-occurrence of words (Lapata, 2003; Soricut and Marcu, 2006), syntactic patterns (Louis and Nenkova, 2012) and discourse relations (Pittel and Nenkova, 2008; Lin et al., 2011). The nature of the tasks addressed by these works (such as determining the correct arrangement order for a set of sentences) makes them focus on learning sequential order of the various discourse components (entities, ideas, etc.). Our goal, instead, is to choose between alternatives of discourse components themselves (and not just their order) to produce a consistent story.

5 Conclusion

Story comprehension is a complex Natural Language Understanding task involving linguistic intelligence as well as a semantic and social knowledge of the real world. This paper studies story comprehension from the perspective of learning what is likely to happen next in a story. We present a model that given a short story, predicts its correct ending. It incorporates three aspects of story-understanding, that are based on an analysis of the events, sentiments and topics described in the story. While this is the best-performing model till date on this task, our analysis indicates a need for even deeper analysis of human behavior and societal norms to further improve our understanding. This work emphasizes that there are multiple aspects to story understanding, which future research can build upon.

Acknowledgement

This work is partly supported by the IBM-ILLINOIS Center for Cognitive Computing Systems Research (C3SR) - a research collaboration as part of the IBM Cognitive Horizon Network; by the US Defense Advanced Research Projects Agency (DARPA) under contract FA8750-13-2-0008; and by the Army Research Laboratory (ARL) under agreement W911NF-09-2-0053. The views expressed are those of the authors and do not reflect the official policy or position of IBM, the Department of Defense or the U.S. Government.

References


