Picking Apart Story Salads

Su Wang[1,2] **Eric Holgate**[1] **Greg Durrett**[3] **Katrin Erk**[1]
[1]Department of Linguistics
[2]Department of Statistics and Data Science
[3]Department of Computer Science
The University of Texas at Austin
shrekwang@utexas.edu holgate@utexas.edu
gdurrett@cs.utexas.edu katrin.erk@mail.utexas.edu

Abstract

During natural disasters and conflicts, information about what happened is often confusing, messy, and distributed across many sources. We would like to be able to automatically identify relevant information and assemble it into coherent narratives of what happened. To make this task accessible to neural models, we introduce *Story Salads*, mixtures of multiple documents that can be generated at scale. By exploiting the Wikipedia hierarchy, we can generate salads that exhibit challenging inference problems. Story salads give rise to a novel, challenging clustering task, where the objective is to group sentences from the same narratives. We demonstrate that simple bag-of-words similarity clustering falls short on this task and that it is necessary to take into account global context and coherence.

1 Introduction

When a natural disaster strikes or a conflict arises, it is often hard to determine what happened. Information is messy and confusing, spread out over many messages, buried in irrelevant text, and even conflicting. For example, when flight MH-17 crashed in Ukraine in 2014, there were initially many theories of what happened, including a missile strike initiated by Russia-affiliated militants, a missile strike by the Ukrainian military, and a terrorist attack. There was no single coherent interpretation of what happened, but multiple, messy narratives, a *story salad*. We would like to be able to automatically identify relevant information and assemble it into coherent narratives of what happened. This task is also the subject of an upcoming task at the Text Analysis Conference.[1]

Picking apart a story salad is a hard task that could in principle make use of arbitrary amounts of inference. But it is also a task in which coherence judgments could play a large role, the simplest being topical coherence, but also narrative coherence (Chambers and Jurafsky, 2008, 2009; Pichotta and Mooney, 2016; Mostafazadeh et al., 2017), overall textual coherence (Barzilay and Lapata, 2008; Logeswaran et al., 2018), and coherence in the description of entities. This makes it an attractive task for neural models.

(A) Some of the prisoners were survivors of the Battle of Qala-i-Jangi in Mazar-i-Sharif. (A) Chechnya came under the influence of warlords. (B) The U.S. invaded Afghanistan the same year when several Taliban prisoners were shot. (A) Russian federal troops entered Chechnya and ended its independence. (A) The Russian casualties included at least two commandos killed and 11 wounded. (B) The dead were buried in the same grave under the authority of Commander Kamal.

Figure 1: A story salad involving two articles, about a Russian military operation in Chechnya (A) and about a U.S. operation in Afghanistan (B). These two articles are topically similar but their mixture can still be disentangled based on narrative coherence.

To make the task accessible to neural models, we propose a simple method for creating simulated story salad data at scale: we mix together sentences from different documents. Figure 1 shows an example mixture of two articles from Wikipedia, one on the Russia-Chechnya conflict and one on a conflict between the U.S. and Afghanistan. By controlling how similar the source documents are, we can flexibly adjust the difficulty of the task. In particular, as we show below, we can generate data that exhibits challenging inference problems by exploiting the Wikipedia category structure.[2] While this data is still simpler than story salads arising naturally, it approximates the task, is sufficiently challenging for modeling, and can be generated in large amounts.[3]

[1] https://tac.nist.gov/2018/SM-KBP/index.html

[2] In particular, while we do not focus on creating mixtures with conflicting information, it can often be found in mixtures created based on Wikipedia categories, as shown in Figure 2.

[3] The story salad data is available at http://www.katrinerk.com/home/software-and-data/picking-apart-story-salads-1. The Wikipedia

Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing, pages 1455–1465
Brussels, Belgium, October 31 - November 4, 2018. ©2018 Association for Computational Linguistics

We explore some initial models for our Story Salad task. As the aim of the task is to group story pieces into stories, we start with straightforward clustering based on topic similarity. But topic similarity is clearly not enough to group the right pieces together. For example, the two articles in Figure 1 are both about armed conflicts, but the Russia-Chechnya sentences in the example form a group *in contrast to* the U.S.-Afghanistan sentences. To model this, we learn sentence embeddings adapted to the clustering task and with access to global information about the salad at hand. We also test an extension where to decide whether to group two sentences together, the model mutually attends to the sentences during encoding in order to better focus on the commonalities and differences of these two sentences. Both extensions lead to better models (6-13% improvement in accuracy with a model incorporating both), confirming that the task requires more than just general topical similarity. But there is much room for improvement, in particular on salads generated to be more difficult, where performance is around 15 points lower than on arbitrary mixtures.

2 Related Work

Building on early work in *script learning* (Schank and Abelson, 1977), Chambers and Jurafsky (2008) introduce *narrative schema* and propose the "narrative cloze" task where the modeling objective is to predict the event happening next. The topic has since seen many extensions and variants coupled with increasingly sophisticated models (Chambers and Jurafsky, 2009) including neural networks (Granroth-Wilding and Clark, 2016; Pichotta and Mooney, 2016; Mostafazadeh et al., 2017). This line of work is related to story salads in that our aim of separating entangled narratives in a document mixture also leverages within-narrative coherence. Our work, however, is very different from narrative cloze: (i) we group sentences/events rather than predicting what happens next; (ii) crucially, the narrative coherence in story salads is *in context*, in that a narrative clustering is only meaningful with respect to a particular document mixture (see Section 5, 6), while in narrative cloze the next event is predicted on a "global" level.[4]

Working with labeled story salad examples, we draw inspiration from previous work on supervised clustering (Bilenko et al., 2004; Finley and Joachim, 2005). We also take advantage of the recent success of deep learning in leveraging a continuous semantic space (Pennington et al., 2014; Kiros et al., 2015; Mekala et al., 2017; Wieting and Gimpel, 2017; Wieting et al., 2017) for word/sentence/event encoding; neural components for enhanced supervised clustering (Bilenko et al., 2004), in particular LSTMs (Hochreiter and Schmidhuber, 1997; Dai and Le, 2015), CNNs (Kim, 2014; Conneau et al., 2017), and attention mechanisms (Bahdanau et al., 2015; Hermann et al., 2015; Lin et al., 2017). By exploring our ability to pick apart story salads with these state-of-the-art NLP modeling tools, we attempt to (i) show the value of the story salad task as a new NLP task that warrants extensive research; (ii) understand the nature of the task and the challenges it sets forth for NLP research in general.

The task of picking apart story salads is related to the task of conversation disentanglement (Elsner and Charniak, 2008; Wang and Oard, 2009; Jiang et al., 2018), which is a clustering task of dividing a transcript into a set of distinct conversations. While superficially similar to our Story Salad task, conversation disentanglement focuses on dialogues and has many types of metadata available, such as time stamps, discourse information, and chat handles. Existing systems draw heavily on this metadata. Another related task is the distinction of on-topic and off-topic documents (Bekkerman and Crammer, 2008), which is defined in terms of topical relatedness. In comparison, the story salad task offers opportunities for more in-depth reasoning, as we show below.

3 Data

Natural story salads arise when multiple messy narratives exist to describe the same event or outcome. Often this is because each contribution to the explanation only addresses a small aspect of the larger picture. We can directly simulate the confusion this kind of discourse creates by taking multiple narratives, cutting them into small pieces, and mixing them together.

salads are available for download directly and we have provided code to reconstruct the NYT salads from English Gigaword 5 (available as LDC2003T05).

[4] The story salad task is more similar to multichoice narrative cloze (Granroth-Wilding and Clark, 2016) in this regard, but formulated categorically differently.

Dataset	Salads	Total Words	μ Words/Salad	cos (test)
NYT	573,681	217,841,716	379.726	0.33
NYT-HARD	1,000	20,149	438.220	0.56
WIKI	500,000	197,175,135	394.350	0.35
WIKI-HARD	50,374	21,266,243	422.167	0.64

Table 1: Statistics of the datasets we present. The average topic cosine similarity scores (cos) between the two narratives in document mixtures are computed from the test sets. The NYT, WIKI and WIKI-HARD salads are divided into 80%/20% train/test splits, while the smaller NYT-HARD is treated entirely as test.

Data generation. Story salads are generated by combining content from source documents and randomizing the sentence order of the resulting mixture. In order to ensure appropriately sized salads, we require that each source document contain at least eight sentences. Furthermore, to avoid problematically large salads, we pull paragraphs from source documents one at a time until the eight sentence minimum is met. While this procedure can be used to mix any number of documents, we currently present mixtures of two documents.

We utilize two different corpora as sources for story salad generation: (i) the subset of New York Times articles presented within English Gigaword (Graff and Cieri, 2003) and (ii) English Wikipedia[5] (Wikipedia contributors, 2004). An overview of the datasets is available in Table 1.

Gigaword. From the New York Times subset of Gigaword, we compiled a set of 573,681 mixtures we call NYT. Each mixture in this set is constructed from source articles pulled from the same month and year. Because this temporal constraint is the only restriction put on what articles can be mixed, it is possible for a salad to be constructed from topically disparate source documents (e.g., a restaurant review and a political story). We intend NYT to be relatively easy on the whole as a result of this design choice.

However, it is also possible for articles about dominant news stories and trends (e.g., the OJ Simpson trial in the summer of 1994) to be mixed as a result of the same temporal constraint. We therefore pulled out a curated subset of NYT consisting only of salads generated from highly topically similar source documents which we call NYT-HARD. This subset consists of the 1,000 salads where the source documents are most topically similar. We calculate topic similarity scores by computing the cosine similarity between the average word embeddings for each source document (denoted cos hereafter)

$$\cos(d) = \frac{g(\omega_1) \cdot g(\omega_2)}{\parallel g(\omega_1) \parallel \parallel g(\omega_2) \parallel} \tag{1}$$

where ω_1 and ω_2 are the source documents, g is a function that computes the average word embedding of a document, and d is the salad under evaluation. The cos scores on the test portion of the datasets are presented in Table 1.

Wikipedia. From Wikipedia, we present an additional set of 500k salads constructed by combining random articles which we call WIKI.

We also leverage Wikipedia category membership as a form of human-annotated topic information. We use this to create a set of 50,374 salads, henceforth called WIKI-HARD, by restricting the domain of articles to only those appearing in categories containing the words *conflict* and *war*. Each mixture in this set is generated from source articles from the same category in order to produce highly difficult mixtures. We intend this to be a challenge set in this domain as the constituent articles for a given mixture are intentionally selected to be closely related. While we have used the category information to construct an intentionally very difficult set for this paper, we note that this procedure can be used to create sets of varying difficulty.

The fact that WIKI-HARD is generated from human-annotated category labels differentiates it from NYT-HARD in the source of its difficulty. After manually reviewing 20 samples from each *-HARD dataset, we found that NYT-HARD more frequently contains salads that are impossible for humans to pick apart while WIKI-HARD more frequently contains salads that are possible, though challenging. In particular, in 9 out of 20 WIKI-HARD salads we found that access to world knowledge and inference would be beneficial. Nevertheless, the two *-HARD datasets are both high in topic similarity (Table 1).

In Figure 2 we present sentences from a sample WIKI-HARD salad that can be solved with world knowledge. In this salad, we learn about two individuals. We can tell that Randle, born in 1855, is unlikely to also have been enrolled in high school in 1913 at the age of 58. We also learn that Randle was a doctor, while Martins, the other individual, was involved in theater. From this, we can deduce that the individual who "also worked as a wrestler" is more likely to be Martins than Randle.

[5] Wikipedia dump pulled on January 20, 2018.

(A) John K. Randle was born on 1 February 1855, son of Thomas Randle, a liberated slave from an Oyo village in the west of what is now Nigeria. (B) In 1913 he was enrolled in Eko Boys High School but dropped out. (B) Martins joined the theatre and from there took on various theatre jobs to survive. (A) Born in Sierra Leone , he was one of the first West Africans to qualify as a doctor in the United Kingdom. (B) He also worked as a wrestler (known as "Black Butcher Johnson").

Figure 2: A story salad from WIKI-HARD, sourced from articles belonging to the *Nigerian people of World War I* category. The sentences from this salad have been rearranged for clearer presentation.

Event Representation. Finally, we explore a form of document representation that has been shown to be useful in narrative schema learning, a related task. We include variants of NYT and NYT-HARD with story salads consisting of event tuple representations instead of natural language sentence representations, as in Pichotta and Mooney (2016). We label these variants as NYT-EVENT and NYT-EVENT-HARD. Event tuples are in the form <VERB, SUBJ, DOBJ, PREP, POBJ>, where as many preposition and prepositional object pairs as necessary are allowed.[6]

Summary. The story salads we present here are, in the end, simpler than those that occur naturally in the news or on social media: for one thing, sentences drawn from a document written by a single author should exhibit a high degree of coherence. We have also shown that we can use Wikipedia category annotations to produce large-scale story salad datasets with customizable levels of difficulty, enabling us to increase the difficulty of the task as performance increases. In the following section, we see that both our standard and *-HARD mixtures are challenging for current models. Furthermore, our WIKI-HARD dataset contains salads featuring conflicting information and is an attractive setting for building models with deeper reasoning capabilities.

4 Models

We treat the story salad task as a narrative clustering task where, in our dataset, each salad is comprised of two clusters. Accordingly, the first baselines we consider are standard clustering approaches.

Baselines. Our first baseline is a simple *uniform baseline* (hereafter **UNIF**), where we assign all sentences in a document mixture to a single cluster. Under UNIF the clustering accuracy is the percentage of the majority-cluster sentences, e.g. if a mixture has 7 sentences from one narrative and 3 from the other, then the accuracy is 0.7.

Additionally, we explore a family of baselines that consist of clustering off-the-shelf sentence embeddings. We choose k-medoids[7] (hereafter **KM**) as our clustering algorithm. For sentence embeddings, we experimented with (i) averaged 300D GloVe embeddings (Pennington et al., 2014), which have been shown to produce surprisingly strong performance in a variety of text classification tasks (Iyyer et al., 2015; Coates and Bollegala, 2018); (ii) skip-thought embeddings (Kiros et al., 2015); and (iii) SCDV (Mekala et al., 2017), a multisense-aware sentence embedding algorithm which builds upon pretrained GloVe embeddings using a Gaussian mixture model. Averaged GloVe embeddings gave the best performance in our experiments; to avoid clutter, we only report those results henceforth.

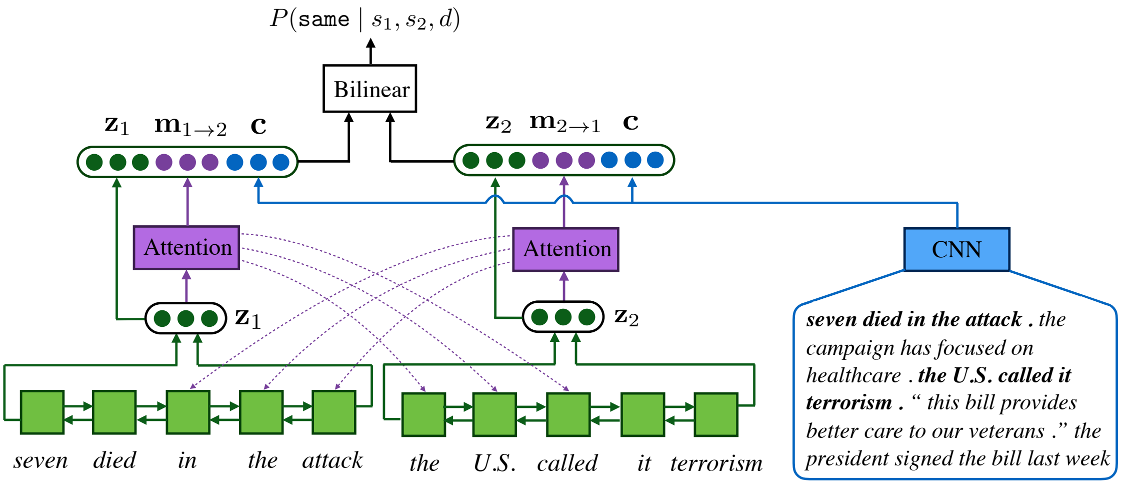
Neural supervised clustering. Our baselines work directly on sentence embeddings and therefore ignore the task-specific supervision available in our labeled training data. Inspired by the work in Bilenko et al. (2004) and Finley and Joachim (2005) on supervised clustering, we aim to exploit this supervision using a learned distance metric in our clustering.[8]

Figure 3 shows our model, which produces a distribution $P(\texttt{same} \mid s_1, s_2, d)$: the probability that two sentences s_1 and s_2 taken from document mixture d are in the same cluster. We train this model as a binary classifier on sampled pairs of sentences to distinguish same-narrative sentence pairs (positive examples) from different-narrative pairs (negative examples). $1 - P(\texttt{same} \mid s_1, s_2, d)$ is then used by the clusterer as the pairwise distance metric. Given the pairwise distance between all sentence pairs in a mixture, the KM algorithm

[6] Event tuples are extracted via the extractor presented in Cheng and Erk (2018), and copular verbs are not treated as events, meaning that some sentences translate to null events.

[7] K-medoids is chosen as a substitute for k-means because the latter does not extend easily to our classifier-aided neural models: it does not work when only pairwise distances are available. In empirical evaluation we found k-means and k-medoids to produce very similar accuracy scores when using off-the-shelf embeddings. Experiments with hierarchical agglomerative clustering (not reported here) showed it to perform worse than either method.

[8] In early experiments, another strong candidate we tried is a joint model of a sentence autoencoder and a clustering algorithm (Yang et al., 2017). However, this produces subpar performance (weaker than the strongest baseline), due partially to scalability issues in learning these jointly.



$P(\texttt{same} \mid s_1, s_2, d)$
Bilinear
$\mathbf{z}_1$ $\mathbf{m}_{1\to 2}$ $\mathbf{c}$
$\mathbf{z}_2$ $\mathbf{m}_{2\to 1}$ $\mathbf{c}$
Attention
Attention
CNN
$\mathbf{z}_1$
$\mathbf{z}_2$
seven died in the attack
the U.S. called it terrorism

seven died in the attack . *the campaign has focused on healthcare .* ***the U.S. called it terrorism .*** *" this bill provides better care to our veterans ." the president signed the bill last week*

Figure 3: BiLSTM sentence pair classifier to determine whether s_1 and s_2 are from the same narrative, augmented with a mutual attention and a context reader. The three subcomponents — the BiLSTM, the mutual attention mechanism, and the context reader — each produce vectors, denoted as $\boldsymbol{z}, \boldsymbol{m}, \boldsymbol{c}$ respectively. In the basic BILSTM model, only $\boldsymbol{z}$ is fed to the bilinear layer (Eq. 2), while more sophisticated models incorporate the additional mutual attention and context vectors.

can then be applied to cluster sentences into two narratives.

Our classifier is a neural network model built on top of LSTM sentence encoders, which perform well at similar text classification tasks (Dai and Le, 2015; Liu et al., 2016).[9] Denoting a sentence as the list of embeddings of its constituent words: $s = \{\boldsymbol{w}_1, \ldots, \boldsymbol{w}_M\}$, we first encode it as a sentence embedding $\boldsymbol{z}$ with a bidirectional LSTM $\boldsymbol{z} = \texttt{BiLSTM}(s)$ and then compute the probability score with a bilinear layer:

$$P(\texttt{same} \mid s_1, s_2) = \sigma(\boldsymbol{z}_1^T W \boldsymbol{z}_2) \qquad (2)$$

This model corresponds to the green subset of Figure 3.

Stronger models. There are two additional effects we might want our model to capture. First, whether two sentences are from the same narrative cannot be determined globally: there aren't two "globally-contrasted"[10] narratives (or bag-of-words based topics) from which sentences are sampled. In other words, sentences are always (pairwise) compared *in the context* of the document mixture from which they are drawn. Second, we want to capture more in-depth interactions between sentences: our sentence embedding scheme for a sentence s_1 should exploit its point of comparison s_2 and encode s_1 with a view of similarities to and differences with s_2. This type of technique has been useful in tasks like natural language inference (NLI) (Bowman et al., 2015; Peters et al., 2018).

To improve contextualization, we add a CNN-based context encoder to the BiLSTM classifier: the reader embeds the whole document salad at hand into a vector. Formally, we compute $\boldsymbol{c} = \text{CNN}(d)$, where in this case CNN denotes a single convolution layer with max pooling in the style of Kim (2014) and d is the concatenation of all sentences in the mixture. This component is shown in blue in Figure 3. The context vector $\boldsymbol{c}$ is then appended to $\boldsymbol{z}$ and fed into the bilinear layer.

To capture the interaction between two sentences in a pair, we employ a *mutual attention* mechanism, which is similar to the attentive reader (Hermann et al., 2015). Let $\boldsymbol{e}_{i,1\ldots n}$ denote the BiLSTM outputs for the tokens of sentence i. Given the encoding $\boldsymbol{z}_1$ of sentence s_1, we compute attention weights and a representation of s_2 as follows:

$$\boldsymbol{\alpha}_{1\to 2} = \text{softmax}_j(\boldsymbol{z}_1^\top \boldsymbol{e}_{2,j})$$
$$\boldsymbol{m}_{1\to 2} = \sum_j \alpha_{1\to 2,j}\, \boldsymbol{e}_{2,j}$$

We compute $\boldsymbol{m}_{2\to 1}$ analogously. This process is shown in purple in Figure 3. The $\boldsymbol{m}$ vectors are used as additional inputs to the bilinear layer.

For comprehensive ablation, we experiment with four variants of neural classifiers: (i) BiL-

[9] Experiments with convolutional encoders here yielded somewhat worse results.

[10] Two stories may be on the same topic and still form clearly different narratives. For example, both narratives in Figure 1 are regarding military conflict.

STM alone (**BILSTM**); (ii) BiLSTM + mutual attention (**BILSTM-MT**); (iii) BiLSTM + context (**BILSTM-CTX**); and (iv) BiLSTM + mutual attention and context (**BILSTM-MT-CTX**).

Event-based models. For the event-based variants of the datasets, NYT-EVENT and NYT-EVENT-HARD, we build three models: (i) FFNN-BILSTM: we input a sentence as a sequence of event embeddings rather than word embeddings as in BILSTM, where a feedforward layer maps the words in an event tuple to an event embedding; (ii) FFNN-BILSTM-MT-CTX: replacing the base BILSTM in (i) with our best model which is enhanced with mutual attention and contextualization; (iii) FFNN-BILSTM-MT-CTX-PRETRAIN: a variant of (ii) that is based on the event embedding pretraining method[11] described in Weber et al. (2018), where events are encoded with a feedforward net (same as (i)) and trained with a word2Vec-like objective, encouraging events that co-occur in the same narrative to have more similar embeddings.

5 Experiments and Analysis

Experimental setup. To stave off sparsity, we impose a vocabulary cut by using only the 100k most frequent lemmas. To evaluate on NYT, NYT-EVENT, WIKI and WIKI-HARD, we sample 20k unique salads (from their respective test portions[12]) to use for both the sentence and event versions of the experiments. For WIKI-HARD, the training combines the training portions of both WIKI and WIKI-HARD. For NYT-HARD, we train on the training portion of NYT and evaluate on NYT-HARD in full as a test set.

All the neural components are constructed with TensorFlow and use the same hyperparameters across variants: a 2-layer BiLSTM, learning rate 1e-5 with Adam (Kingma and Ba, 2014), dropout (Srivastava et al., 2014) rate 0.3 on all layers, and Xavier initialization (Glorot and Bengio, 2010). To create training pairs for the neural classifiers, we randomly sample sentence pairs balanced between same-narrative and different-narrative pairs. We train with a batch size of 32 and stop when an epoch yields less than 0.001% accuracy improvement on the validation set, which is 5% of mixtures sampled from the training data beforehand (the models are not trained on the validation sample). For KM we use the default configurations of off-the-shelf software.[13]

Model	NYT	WIKI	NYT-HARD	WIKI-HARD
UNIF	52.7	50.9	52.5	51.2
KM	76.4	74.9	59.8	60.4
BILSTM	78.5	76.2	55.3	59.8
BILSTM-MT	80.8	78.8	56.7	61.3
BILSTM-CTX	82.6	78.9	63.8	63.7
BILSTM-MT-CTX	**84.9**	**81.8**	**68.0**	**66.6**

Table 2: Clustering accuracy (CA) results from the sentence based experiments. More sophisticated models do better across all datasets, particularly on *-HARD tasks, which are substantially more challenging.

Evaluation. We evaluate all models in terms of a *clustering accuracy* metric (hereafter CA), which is a simple extension from the conventional accuracy metric: we calculate the ratio of correctly clustered sentences in a document mixture, averaged over test mixtures. Given a document mixture d_i, we call its component documents A and B. Let $\texttt{pred}$ be a function that does the clustering by mapping each sentence $s_{i,n}$ of mixture d_i to either A or B, and $\texttt{true}_{AB}$ a function that returns the original pseudo-labels (i.e. $\{A, B\}$) as they are, and $\texttt{true}_{BA}$ flips the pseudo-labels, i.e. $A \to B$ and $B \to A$. Then the clustering accuracy for document d_i by $\texttt{pred}$ is

$$\begin{aligned}\text{CA}(d_i, \texttt{pred}, \texttt{true}) = \max\{&\text{C}(d_i, \texttt{pred}, \texttt{true}_{AB}),\\ &\text{C}(d_i, \texttt{pred}, \texttt{true}_{BA})\}\\ \text{C}(d_i, \texttt{pred}, \texttt{true}) =&\\ \frac{1}{N_i}\sum_n &\mathbb{1}[\texttt{pred}(s_{i,n}) = \texttt{true}(s_{i,n})]\end{aligned}$$

where $s_{i,n}$ is the n-th sentence of mixture d_i.

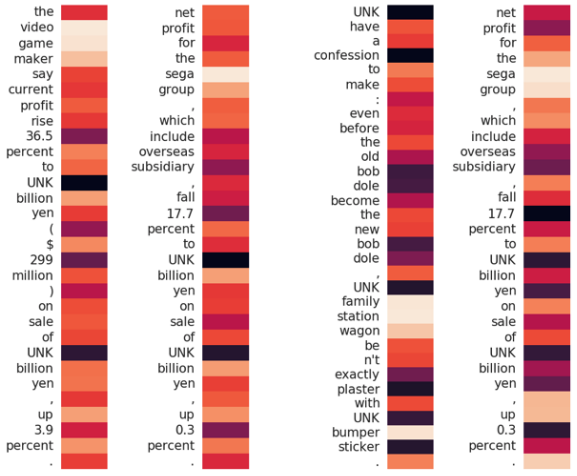
Sentence based models. First we evaluate the sentence based models. We first run the UNIF baseline on all our datasets, where we obtain near-50% clustering accuracy. This indicates that the data are all balanced in the number of sentences in the two narratives of the mixtures. We then run k-medoids (KM) on sentence embeddings as a baseline to compare to the classifier-aided models. The results are summarized in Table 2.

We first observe that the KM is a strong baseline and outperforms the supervised BILSTM system in the harder settings. Adding the mutual attention mechanism and contextualization, however, im-

[11] In Weber et al. (2018), a more complex tensor-based model is applied. Using exactly same method in our experiments we obtain weaker results.

[12] Test sets available with data release.

[13] github.com/letiantian/kmedoids



(a) from same narrative

(b) from different narratives

Figure 4: Attention weight heatmaps for a random sample with BILSTM-MT-CTX. Lighter color indicates higher attention weights. The two heatmaps in the same block are for the attention weights of one sentence attending to the other. In (a), we see related concepts being identified (*video game* and *sega*), while in (b), we see a contrast (*family station wagon* and *sega group*).

Type	Model	NYT	WIKI
-CONTEXT	BILSTM	-0.40^*	-0.43^*
	BILSTM-MT	-0.38^*	-0.40^*
+CONTEXT	BILSTM-CTX	-0.31^*	-0.30^*
	BILSTM-MT-CTX	-0.27^*	-0.25^*

Table 3: Spearman's ρ correlation between clustering accuracy (CA) and topic similarity (COS) in the evaluation with NYT and WIKI. The p-values are all below 0.01 (indicated by *). Contextualized models (+CONTEXT) are more robust to high topic similarity than their uncontextualized counterparts (-CONTEXT), indicated by the lower negative correlation between their accuracy and topic similarity.

prove BILSTM substantially. In addition, the performance boost from the two components seems more or less orthogonal, as shown by the much stronger accuracy of the combined model (i.e. BILSTM-MT-CTX) than the models with a single component (i.e. BILSTM-MT and BILSTM-CTX). Overall, the large margin of KM and classifier-aided models above the UNIF baseline indicates that separating story salads is a valid task where generalizable patterns can be exploited by machine learning techniques.

Why would the mutual attention mechanism help? Plotting the attention weights of randomly selected samples, we see distributionally similar words being attended to in Figure 4a. Intuitively, a BiLSTM compresses a sentence into a single vector, leading to information loss (Conneau et al., 2018). Mutual attention enriches this representation by allowing us access to detailed information in sentences at word-level resolution by capturing lexical similarity. Even more interestingly, we observe a synergistic effect between mutual attention and contextualization: with the context reader added, we see high attention weights on words/phrases which bear little distributional similarity but are important for connecting/contrasting different narratives. For example, in Figure 4b, *sega group* and *family station wagon* are selected by the attention, despite not having similar words in the other sentences. These words are crucial in identifying the two narratives in this mixture: one is about a Japanese video game company, the other is on vehicle manufacturing in the U.S.

Another observation is that all models see drastic reduction in accuracy in the *-HARD version of the data. In fact, the clustering accuracy corresponds well with our topic similarity metric (COS, Eq. 1; Table 1) across models. In addition, COS is negatively correlated with clustering accuracy for all mixtures (Table 3).

From the results we also see that contextualization brings clear performance improvement. This supports our hypothesis that the Story Salad task is a nonstandard clustering task where the contrast of two narratives is only meaningful *in the context* of the particular mixture where they reside, rather than on a corpus-general level. Taking the example in Figure 1, the Russian-Chechnya and the U.S.-Afghanistan narratives are contrasted in that mixture, but one can easily imagine a mixture where they are in the same narrative and are contrasted to another narrative on business affairs. Further, contextualized models are less vulnerable to the performance reduction on mixtures with high topic similarity: for one thing, contextualization improves performance over the base BILSTM on both regular and *-HARD datasets. Secondly, computing the correlation between clustering accuracy and topic similarity, we see a lower negative correlation for contextualized models, true for both NYT and WIKI datasets (Table 3).

Event based models. While the accuracy scores in the event based experiments are in general lower than those in the sentence based (Table 4), overall we observe the same pattern that

Model	NYT-EVENT	NYT-EVENT-HARD
KM	64.7	55.3
FFNN-BILSTM	64.9	54.8
*-MT-CTX	66.8	59.1
*-MT-CTX-PRETRAIN	**70.2**	**61.4**

Table 4: Clustering accuracy (CA) results from the event based experiments. *-MT-CTX is a short hand for FFNN-BILSTM-MT-CTX. The same notation applies for the following models.

mutual attention and contextualization contribute substantially to the performance. More interestingly, the performance reduction on the topically highly similar *-HARD is more mild compared to the sentence based experiments, which provides initial evidence that event-based narrative encoding allows the models to be more robust to distraction by lexical overlap in topically similar narratives. Finally we see that the event pretraining with Weber et al. (2018)'s technique brings additional improvement over a contextualized system.

The results open up a door for future work: (i) our simple models do not make use of coreference, narrative schema or world knowledge, which are intuitively promising components to introduce (see, e.g., the salad in Figure 2); (ii) more sophisticated model architectures may help capture the information missed by our models: moving from the sentence version to the event version, we lose many words which may have provided crucial cues in the sentence-based experiments.

Error analysis. In order to understand the errors made by each model, we performed a manual analysis of a small sample of bad clusterings. In a sample of 60 mixtures from NYT (test set), we considered all clusterings for which accuracy was less than 0.65. Among the 60 mixtures, the base model had an accuracy this low for 27 mixtures, the BILSTM-MT and BILSTM-CTX model had 13 low-accuracy mixtures each, and BILSTM-MT-CTX had 3. Each mixture was manually annotated by 2 annotators as being sourced from (i) thematically closely related documents (e.g., two stories on the same political event), (ii) thematically distinct documents (e.g., a political story and a sports story), or (iii) cannot tell.

Our analysis showed that the base BILSTM model has difficulty even in cases where the source documents for the salad are thematically distinct. This was the case in 9 of 27 bad clusterings. The BILSTM-MT, BILSTM-CTX and BILSTM-MT-CTX models not only have many fewer bad clusterings, they also show low accuracy almost exclusively in mixtures of related documents (2 cases of distinct documents for BILSTM-CTX, none for BILSTM-MT or BILSTM-MT-CTX). Figure 5 shows an example of a bad clustering of two unrelated documents, produced by the base BILSTM model.

(A) lehman brothers be one of several investment bank eager to get UNK hand on state asset, across the nation and in massachusetts (A) former massachusetts governor william f. weld, a staunch supporter of privatization during UNK administration, have UNK in the hall of the state house, now as a corporate lawyer try to drum up support for the sale of lucrative state asset. (B) officially, the rays option dukes, 22, to class a vero beach and place UNK on the temporary inactive list, where UNK will remain for an undetermined amount of time as UNK undergo counseling. (B) UNK apparently will receive UNK $ 380,000 major-league salary .

Figure 5: An example of (preprocessed) sentences from two unrelated documents being that have been clustered into a single cluster by the base model. Document (A) is an article about proposed privatization of public assets, while Document (B) is an article about happenings in Major League Baseball.

In a second study, we rated the same 60 samples by their difficulty for a human, focusing in particular on mixtures that went from low performance (0.5-0.65) in the BILSTM model to high performance (0.8-1.0) in another model. For BILSTM-CTX we find that only 2 out of 11 mixtures with such marked improvement over BILSTM were hard for humans; for BILSTM-MT only 1 out of 9 markedly improved mixtures was hard for humans. But for BILSTM-MT-CTX, 8 out of 17 markedly improved mixtures were hard for humans, indicating that more sophisticated models do better not only on easy but also on hard cases.

In a third small study, we compare NYT-HARD and WIKI-HARD for their difficulty for humans, looking at 20 mixtures each. Here, very interestingly, we find more mixtures that are impossible for humans in NYT-HARD (10 cases, example in Figure 6) than WIKI-HARD (3 cases). This presents a clear discrepancy between difficulty for humans and difficulty for models: the models do better on NYT-HARD which is harder for us. While we would not want to draw strong conclusions from a small sample, this hints at possibilities of future work where world knowledge, which is likely to be orthogonal to the information picked up by the models, can be introduced to improve performance (e.g. Wang et al. (2018)).

Note that unlike many other NLP tasks where

(A) The most basic question face the country on energy be how to keep supply and demand in line. The Democrats would say : "what can UNK do to make good use of what UNK have get?" (B) Oil price dominate the 31-minute news conference, hold here near pittsburgh. (B) Vice President Al Gore hold UNK first news conference in 67 day on Friday, defend UNK call for the release of oil from the government's stockpile and and vow that UNK would "confront friend and foe alike" over the marketing of violent entertainment to child, despite the million in donation UNK receive from Hollywood. (A) With oil price up, consumer agitate and the winter heating season loom, Vice President Al Gore and Gov. George W. Bush be go at UNK on energy policy, seek to draw sharp distinction over an issue on which both candidate have political vulnerability. (B) On other topic, Gore say UNK be not nervous about the upcoming debate, but be incredulous when a reporter ask whether UNK be confident UNK have the election lock up. (A) Bush, who criticize the decision as a political ploy to drive down price just ahead of election day, be schedule to discuss energy policy in a speech on Friday. (B) On Friday, Bush call Gore a "flip-flopper", say UNK proposal to tap into the reserve be a political ploy.

Figure 6: Part of a story salad that is impossible for a human to pick apart (source: NYT-HARD). "UNK" represents out-of-vocabulary tokens, and all the words are lemmatized. Both narratives, i.e. (A) and (B) involve the characters Al Gore and George Bush, and both are on the topic of energy, with strongly overlapping vocabulary.

human performance sets the ceiling for the best achievable results (e.g. span-prediction based question answering (Rajpurkar et al., 2016), where all the information needed for the correct answer is available in the input), successfully picking apart narratives in a story salad may require consulting an external knowledge base, which affords machine learning models a clear advantage over humans. For example, recognizing that *Commander Kamal* is likely to be Afghani based on his name, which is not knowledge every reader possesses, would allow us to successfully cluster the sentence with the U.S.-Afghanistan narrative rather than the Russian-Chechnya narrative.

6 Conclusion

We have presented a technique to generate *Story Salads*, mixtures of multiple narratives, at scale. We have demonstrated that the difficulty of these mixtures can be manipulated either based on document similarity or based on human-created document categories. This data gives rise to a challenging binary clustering task (but easily extended to n-ary), where the aim is to group sentences that come from the same original narrative. As coherence plays an important role in this task, the task is related to work on narrative schemas (Chambers and Jurafsky, 2008; Pichotta and Mooney, 2016) and textual coherence (Barzilay and Lapata, 2008; Logeswaran et al., 2018). The automated and scalable data generation technique allows for the use of neural models, which need large amounts of training data.

Conducting a series of preliminary experiments on the data with common unsupervised clustering algorithms (Cao and Yang, 2010) and variants of neural network-based (Kim, 2014; Dai and Le, 2015; Liu et al., 2016) supervised clustering (Bilenko et al., 2004; Finley and Joachim, 2005) models, we have (i) verified the validity of the task where generalizable patterns can be learned through machine learning techniques; (ii) shown that this is a nonstandard clustering task in which the contrast between narratives is *in context* as opposed to global; (iii) found that there is a class of mixtures that are doable for humans but very difficult for our current models, and that in particular the category-based method creates a high proportion of such mixtures.

Our work opens up a large number of directions for future research. First, while our models obtain strong results on simpler story salads, they have low performance on more difficult mixtures with high topical similarity. Second, there are many intuitively promising sources of information that we have not explored, such as coreference. And third, our models rely on pairwise similarity-based coherence learning, which leads to the natural question of whether structured prediction would improve performance.

Acknowledgments

This research was supported by the DARPA AIDA program under AFRL grant FA8750-18-2-0017. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the authors and do not necessarily reflect the view of DARPA, DoD or the US government. We acknowledge the Texas Advanced Computing Center for providing grid resources that contributed to these results. We are grateful to the anonymous reviewers for helpful discussions.

References

Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. 2015. Neural Machine Translation by Jointly Learning to Align and Translate. In *Proceedings of NIPS*.

Regina Barzilay and Mirella Lapata. 2008. Modeling Local Coherence: An Entity-Based Approach. *Computational Linguistics*, 34(1):1–34.

Ron Bekkerman and Koby Crammer. 2008. One-class Clustering in the Text Domain. In *Proceedings of EMNLP*, pages 41–50.

Mikhail Bilenko, Sugato Basu, and Raymond J. Mooney. 2004. Integrating Constraints and Metric Learning in Semi-Supervised Clustering. In *Proceedings of ICML*.

Samuel R. Bowman, Gabor Angeli, Christopher Potts, and Christopher D. Manning. 2015. A Large Annotated Corpus for Learning Natural Language Inference. In *Proceedings of EMNLP*.

Danyang Cao and Bingru Yang. 2010. An Improved K-Medoids Clustering Algorithm. In *Proceedings of ICCAE, IEEE*.

Nathanael Chambers and Daniel Jurafsky. 2008. Unsupervised Learning of Narrative Event Chains. In *Proceedings of ACL*.

Nathanael Chambers and Daniel Jurafsky. 2009. Unsupervised Learning of Narrative Schemas and Their Participants. In *Proceedings of ACL*.

Pengxiang Cheng and Katrin Erk. 2018. Implicit Argument Prediction with Event Knowledge. In *Proceedings of NAACL*.

Joshua Coates and Danushka Bollegala. 2018. Frustratingly Easy Meta-Embedding Computing Meta-Embeddings by Averaging Source Word Embeddings. In *Proceedings of NAACL*.

Alexis Conneau, German Kruszewski, Guillaume Lample, Loïc Barrault, and Marco Baroni. 2018. What You Can Cram into a Single Vector: Probing Sentence Embeddings for Linguistic Properties. In *Proceedings of ACL*.

Alexis Conneau, Holger Schwenk, Yann LeCun, and Loïc Barrault. 2017. Very Deep Convolutional Networks for Text Classification. In *Proceedings of EACL*.

Andrew M. Dai and Quoc V. Le. 2015. Semi-supervised sequence learning. In *Proceedings of NIPS*.

Micha Elsner and Eugene Charniak. 2008. You Talking to Me? A Corpus and Algorithm for Conversation Disentanglement. In *Proceedings of ACL*.

Thomas Finley and Thorsten Joachim. 2005. Supervised Clustering with Support Vector Machines. In *Proceedings of ICML*.

Xavier Glorot and Yoshua Bengio. 2010. Understanding the Difficulty of Training Deep Feedforward Neural Networks. In *Proceedings of ICML*.

David Graff and Christopher Cieri. 2003. English Gigaword LDC2003T05. *Linguistic Data Consortium*.

Mark Granroth-Wilding and Stephen Clark. 2016. What Happens Next? Event Prediction Using a Compositional Neural Network model. In *Proceedings of AAAI*.

Karl Moritz Hermann, Tomas Kocisky, Edward Grefenstette, Lasse Espeholt, Will Kay, Mustafa Suleyman, and Phil Blunsom. 2015. Teaching Machines to Read and Comprehend. In *Proceedings of NIPS*.

Sepp Hochreiter and Juergen Schmidhuber. 1997. Long Short-Term Memory. *Neural Computation*, 9(8):1735–1780.

Mohit Iyyer, Vrun Manjunatha, Jordan Boyd-Graber, and Hal Daumé III. 2015. Deep Unordered Composition Rivals Syntactic Methods for Text Classification. In *Proceedings of ACL*.

Jyun-Yu Jiang, Francine Chen, Yang-Ying Chen, and Wei Wang. 2018. Learning to Disentangle Interleaved Conversational Threads with a Siamese Hierarchical Network and Similarity Ranking. In *Proceedings of NAACL*.

Yoon Kim. 2014. Convolutional Neural Networks for Sentence Classification. In *Proceedings of EMNLP*.

Diederik P. Kingma and Jimmy Ba. 2014. Adam: a Method for Stochastic Optimization. In *Proceedings of ICLR*.

Ryan Kiros, Yukun Zhu, Ruslan Salakhutdinov, Richard S. Zemel, Antonio Torralba, Raquel Urtasun, and Sanja Fidler. 2015. Skip-Thought Vectors. In *Proceedings of NIPS*.

Zhouhan Lin, Minwei Feng, Cicero Nogueira dos Santos, Mo Yu, Bing Xiang, Bowen Zhou, and Yoshua Bengio. 2017. A Structured Self-attentive Sentence Embedding. In *Proceedings of ICLR*.

Pengfei Liu, Xipeng Qiu, and Xuanjing Huang. 2016. Recurrent Neural Network for Text Classification with Multi-task Learning. In *Proceedings of IJCAI*.

Lajanugen Logeswaran, Honglak Lee, and Dragomir Radev. 2018. Sentence Ordering and Coherence Modeling Using Recurrent Neural Networks. In *Proceedings of AAAI*.

Dheeraj Mekala, Vivek Gupta, Bhargavi Paranjape, and Harish Karnick. 2017. SCDV: Sparse Composite Document Vectors using soft clustering over distributional representations. In *Proceedings of EMNLP*.

Nasrin Mostafazadeh, Michael Roth, Annie Louis, Nathanael Chambers, and James F. Allen. 2017. LSDSem 2017 Shared Task: The Story Cloze Test. In *LSDSem 2017 Shared Task*.

Jeffrey Pennington, Richard Socher, and Christopher D. Manning. 2014. GloVe: Global Vectors for Word Embeddings. In *Proceedings of EMNLP*.

Matthew E. Peters, Mark Neumann, Mohit Iyyer, Matt Gardner, Christopher Clark, Kenton Lee, and Luke Zettlemoyer. 2018. Deep Contextualized Word Representations. In *Proceedings of NAACL*.

Karl Pichotta and Raymond Mooney. 2016. Using Sentence-Level LSTM Language Models for Script Inference. In *Proeedings of ACL*.

Pranav Rajpurkar, Jian Zhang, Konstantin Lopyrev, and Percy Liang. 2016. SQuAD: 100,000+ Questions for Machine Comprehension of Text. In *Proceedings of EMNLP*.

Roger C. Schank and Robert P. Abelson. 1977. Scripts, Plans, Goals and Understanding. *Lawrence Erlbaum*.

Nitish Srivastava, Geoffrey Hinton, Alex Krizhevsky, Ilya Sutskever, and Ruslan Salakhutdinov. 2014. Dropout: a Simple Way to Prevent Neural Network from Overfitting. *JMLR*, 15:1929–1958.

Lidan Wang and Douglas W. Oard. 2009. Context-based Message Expansion for Disentanglement of Interleaved Text Conversation. In *Proceedings of NAACL*.

Su Wang, Greg Durrett, and Katrin Erk. 2018. Modeling Semantic Plausibility by Injecting World Knowledge. In *Proceedings of NAACL*.

Noah Weber, Niranjan Balasubramanian, and Nathanael Chambers. 2018. Event Representations with Tensor-based Compositions. In *Proceedings of AAAI*.

John Wieting and Kevin Gimpel. 2017. Revisiting Recurrent Networks for Paraphrastic Sentence Embeddings. In *Proceedings of ACL*.

John Wieting, Jonathan Mallinson, and Kevin Gimpel. 2017. Learning Paraphrastic Sentence Embeddings from Back-Translated Bitext. In *Proceedings of EMNLP*.

Wikipedia contributors. 2004. Plagiarism — Wikipedia, the free encyclopedia. [Online; accessed 20-January-2018].

Bo Yang, Xiao Fu, Nicholas D. Sidiropoulos, and Mingyi Hong. 2017. Towards K-means-friendly Spaces: Simultaneous Deep Learning and Clustering. In *Proceedings of ICML*.