Encoding World Knowledge in the Evaluation of Local Coherence

Muyu Zhang[1]*, **Vanessa Wei Feng**[2], **Bing Qin**[1], **Graeme Hirst**[2], **Ting Liu**[1] **and Jingwen Huang**[1]
[1]Research Center for Social Computing and Information Retrieval
Harbin Institute of Technology, Harbin, China
[2]Department of Computer Science, University of Toronto, Toronto, ON, Canada
{myzhang,qinb,tliu,jwhuang}@ir.hit.edu.cn
{weifeng,gh}@cs.toronto.edu

Abstract

Previous work on text coherence was primarily based on matching multiple mentions of the same entity in different parts of the text; therefore, it misses the contribution from semantically related but not necessarily coreferential entities (e.g., ***Gates*** and ***Microsoft***). In this paper, we capture such semantic relatedness by leveraging world knowledge (e.g., ***Gates*** *is the person who created* ***Microsoft***), and use two existing evaluation frameworks. First, in the unsupervised framework, we introduce semantic relatedness as an enrichment to the original graph-based model of Guinaudeau and Strube (2013). In addition, we incorporate semantic relatedness as additional features into the popular entity-based model of Barzilay and Lapata (2008). Across both frameworks, our enriched model with semantic relatedness outperforms the original methods, especially on short documents.

1 Introduction

In a well-written document, sentences are organized and presented in a logical and coherent form, which makes the text fluent and easily understood. Therefore, coherence is a fundamental aspect of high text quality, and the evaluation of coherence is a crucial component of many NLP applications, such as essay scoring (Miltsakaki and Kukich, 2004), story generation (McIntyre and Lapata, 2010), and document summarization (Barzilay et al., 2002).

A particularly popular model for evaluating text coherence is the entity-based local coherence model of Barzilay and Lapata (2008) (B&L), which extracts mentions of entities in adjacent sentences, and captures local coherence in terms of the transitions in the grammatical role of each mention. Following this direction, a number of extensions have been proposed (Elsner and Charniak, 2008; Elsner and Charniak, 2011; Lin et al., 2011; Feng et al., 2014), the majority of which focus on enriching the original entity features. An exception is the unsupervised model of Guinaudeau and Strube (2013) (G&S), which converts the document into a graph of sentences, and evaluates the text coherence by computing the average out-degree over the entire graph.

However, despite the apparent success of these methods, they rely merely on matching mentions of the same entity, but neglect the contribution from semantically related but not necessarily coreferential entities. For example, the text in Figure 1a[1] has no common entity in s_2 and s_3. However, the transition between them is perfectly coherent, because there exists close semantic relatedness between two distinct entities, ***Gates*** in s_2 and ***Microsoft*** in s_3, which can be captured by the world knowledge that ***Gates*** *is the person who created* ***Microsoft*** (represented by ***Gates-create-Microsoft***). In fact, the issue of absence of common entities between adjacent sentences is quite prevalent. Analyzing the CoNLL 2012 dataset (Pradhan et al., 2012), we found that 42.34% of the time, adjacent sentences do not share common entities. As a result, methods which rely on strict entity matching would fail on these cases.

*This work was partly done while the first author was visiting University of Toronto.

[1]Based on a news item: http://www.cnbc.com/id/101576926

Human Language Technologies: The 2015 Annual Conference of the North American Chapter of the ACL, pages 1087–1096,
Denver, Colorado, May 31 – June 5, 2015.

s_1: In 1980, [Gates]$_S$ licensed [86-DOS]$_O$ from [Tim Paterson]$_X$ for $50,000, which marketed it as [PC-DOS]$_X$.
s_2: [Gates' smartest move]$_S$ was retaining [ownership of the source code]$_O$ of what he and [Allen]$_X$ would develop as [MS-DOS]$_X$.
s_3: [Microsoft]$_S$ got a [licensing fee]$_O$ every time [IBM]$_S$ sold a [PC]$_O$.

(a) A fragment of news text

	Gates	86-DOS	Paterson	PC-DOS	Move	Ownership	Source	Code	MS-DOS	Microsoft	Fee	Time	IBM	PC
s_1	S	O	X	X	-	-	-	-	-	-	-	-	-	-
s_2	S	-	-	-	S	O	O	O	O	-	-	-	-	-
s_3	-	-	-	-	-	-	-	-	-	S	O	X	S	O

(b) The corresponding entity grid

Figure 1: A news text fragment with its corresponding entity grid constructed following B&L (2008). Although s_2 and s_3 share no entity, their transition is still coherent, because ***Gates*** and ***Microsoft*** are semantically related by the knowledge ***Gates-create-Microsoft***.

We wish to incorporate semantic relatedness between different entities into existing models to tackle the problem described above. In particular, we propose to capture such semantic relatedness between different entities with world knowledge represented as triples, e.g., ***Gates-create-Microsoft***. Given a text to be evaluated, we first retrieve relevant world knowledge from multiple sources. For the unsupervised framework of G&S, we integrate knowledge into the original graph-based document representation, in which sentences are the nodes and edges are formed by shared entities and our world knowledge. Then, we adopt a dynamic programming algorithm to produce a coherence score for the text. For the supervised framework of B&L, we incorporate the world knowledge as a novel set of features into the original entity-based model, and train a model to discriminate different degrees of text coherence.

To evaluate the impact of incorporating semantic relatedness, we conduct experiments on two datasets, each of which resembles a real sub-task in the text coherence modeling: **sentence ordering** and **summary coherence rating**. On both tasks, across two frameworks, supervised and unsupervised, we perform a direct comparison between our enhanced model and the original one. On both tasks, our models are shown to be more powerful than the models relying on entity matching only. Moreover, for sentence ordering, world knowledge is shown to be especially useful on short documents.

2 Background

2.1 Entity-based local coherence modeling

The initial entity-based model was developed by B&L. It is based on the intuition that there exists a canonical order of how entities occur in the text. Therefore, we can model text coherence by measuring how mentions of various entities are distributed within the text. Specifically, for a given document d, an entity grid is constructed in which the rows represent the sentences and the columns represent entities. Each grid cell r_{ij} corresponds to the syntactic role of entity e_j in sentence s_i: subject (S), object (O), other (X), or nothing ($-$). For example, Figure 1b shows the entity grid of the text shown in Figure 1a. If an entity serves multiple syntactic roles in a sentence, its grammatical role is resolved according to the priority order: $S \succ O \succ X \succ -$.

Based on the entity grid representation, a local coherence transition is defined as a sequence $\{S, O, X, -\}^n$, representing the grammatical roles or absence of a particular entity across n adjacent sentences. Then, the document is encoded as a feature vector $\Phi(d) = (p_1(d), p_2(d), \ldots, p_m(d))$, where $p_t(d)$ is the normalized frequency of the transition t in the entity grid, and m is the number of predefined transitions. $p_t(d)$ is computed as the number of occurrences of transition t among all entities in the entity grid, divided by the total number of transitions of the same length. Using this feature encoding, the model is then trained as a preference ranking problem between documents of different degrees of coherence.

2.2 Graph-based local coherence modeling

As mentioned previously, most extensions to the entity-based local coherence model focus on enriching the feature set (Filippova and Strube, 2007; Elsner and Charniak, 2011; Lin et al., 2011; Feng et al., 2014), all of which follow a supervised learning framework. To the best of our knowledge, the only exception is the unsupervised method proposed by

G&S, which transforms the entity grid into a sentence graph and measures text coherence by computing the average out-degree of the graph. For a document d, its entity grid is constructed first, following the method described in Section 2.1. Then, a bipartite graph $G = (V_s, V_e, L, W)$ is constructed, where V_s is the set of nodes representing sentences in the text; V_e is the set of nodes representing entities; L is the set of edges associated with a weight $w \in W$. An edge exists between a sentence s_x and an entity e, if and only if e occurs in s_x. Each edge is further associated with a weight $w(e, s_x)$, determined by the grammatical role of the entity e in sentence s_x: 3 for subject (S), 2 for object (O), 1 for other (X), and 0 for nothing ($-$). Note that graph G consists of both sentence nodes and entity nodes.

Then, G is converted to another graph P, which consists of sentence nodes only, where an edge connects two sentence nodes if and only if at least one entity is shared between these two sentences. In P, the weight of each edge is computed by aggregating the edge weights in the original bipartite graph G:

$$w^{(P)}(s_x, s_y) = \sum_{e \in E_{xy}} w^{(G)}(e, s_x) * w^{(G)}(e, s_y), \quad (1)$$

where E_{xy} is the set of entities shared by two sentences s_x and s_y, and $w^{(G)}(e, s_x)$ is the weight of edge between entity e and sentence s_x as illustrated before. The coherence of the document is thus measured by the average out-degree of graph P.

Although this method is purely unsupervised, it achieves a performance comparable with its supervised counterparts, e.g., B&L. However, since this method still relies on matching multiple mentions of the same entity, it misses the important contribution from those semantically related yet distinct entities, e.g. ***Gates*** and ***Microsoft*** in Figure 1a.

3 Finding relevant world knowledge

To supplement existing models with information derived from semantic relatedness, given a document d to be evaluated, we first retrieve all world knowledge related to d. There are two major issues for this process: (1) **knowledge sources**: where can we obtain this knowledge?, and (2) **knowledge selection**: how do we pinpoint the most relevant ones?

Knowledge sources There are two main kinds of knowledge sources: (1) manually edited knowledge bases, such as YAGO (Hoffart et al., 2013), which consists of about 4 million human-edited instances from on-line encyclopedias such as WikiPedia (Denoyer and Gallinari, 2007) and FreeBase (Bollacker et al., 2008), and (2) automatically constructed knowledge bases, such as Reverb (Fader et al., 2011), which covers about 20 million instances extracted from raw texts. Generally speaking, manually edited knowledge bases have better accuracy but lower coverage, while automatically extracted knowledge bases are the opposite. To seek a good balance, we use both YAGO and Reverb as our knowledge sources. In addition, the automatically constructed knowledge bases can be extracted from raw texts of any domain, which makes our method adaptable. Both sources are presented in triples, $argument_1$-*predicate*-$argument_2$, (e.g., ***Gates-create-Microsoft***), where the two arguments are usually entities and the predicate is the relation between them (Zhang et al., 2014).

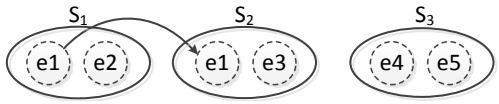
Knowledge selection For each document d, we then select the subset of relevant knowledge instances, in the sense that they represent relations between the entities in d. In particular, we extract all entities in d, and query the knowledge bases to obtain all the knowledge instances in which both of the two arguments, $argument_1$ and $argument_2$, match some of the entities in d.

One issue in knowledge selection is whether to retrieve knowledge instances using exact or partial matching. For a given pair of entities in the text, the chance is rather low to find instances in the knowledge bases where the two arguments perfectly match the pair of entities, because entities in the source document might appear in aliases or abbreviations. In contrast, partial matching between arguments and entities usually increases coverage but at risk of introducing more noise. In our work, to balance accuracy and coverage, when retrieving world knowledge, we use partial matching and form queries only for those entities realized as noun phrases in the text.

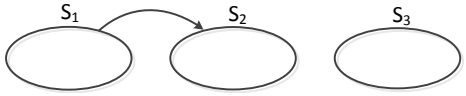
4 World knowledge encoding

4.1 Unsupervised graph-based framework

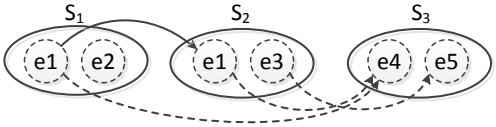
As described in Section 2.2, G&S represents the text as a graph and measures the coherence by the average out-degree of the graph. In this part, we describe



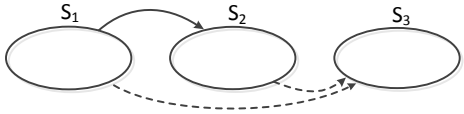
(a) Traditional entity graph relying on entity matching



(b) Traditional sentence graph converted from (a)



(c) Our entity graph encoding relatedness between entities



(d) Our sentence graph converted from (c)

Figure 2: (a) and (b) show the traditional entity and sentence graph based on matching multiple mentions of the same entity; while (c) and (d) represent our entity and sentence graph encoding semantic relatedness between those semantically related but not necessarily coreferential entities (e.g., ***Gates*** and ***Microsoft***) by adding *world-knowledge edges* (dashed lines) accroding to world knowledge (e.g., ***Gates-create-Microsoft***).

how we capture semantic relatedness by encoding world knowledge to the graph-based model. The outline of our method is as follows. Given a document d, we first retrieve relevant world knowledge from multiple sources (see Section 3). Then, we construct an *entity graph* with world knowledge to capture both the distribution information and semantic relatedness between entities (see Section 4.1.1). After that, we convert the entity graph into a *sentence graph* (see Section 4.1.2), in which two sentences are connected not only through common entities but also through world knowledge. Finally, we apply our novel *reachability score computation* **over the sentence graph** to produce a coherence score for the text to be evaluated (see Section 4.1.3).

4.1.1 Entity graph

As shown in Figure 2c, there are two kinds of edges in our *entity graph*: (1) *common-entity edges* (solid lines), which connect different mentions of the same entity, such as ***Gates*** in s_1 and ***Gates*** in s_2; (2) *world-knowledge edges* (dashed lines), which connect different entities through certain world knowledge, such as ***Gates*** in s_2 and ***Microsoft*** in s_3 related by ***Gates-create-Microsoft***. This representation captures not only the distribution information of individual entities but also the semantic relatedness between different entities. In contrast, the original graph-based model by G&S (Figures 2a and 2b) includes common-entity edges only and misses the semantic relatedness information. Formally, for a document d, we define its entity graph as $G = (V, L_m, L_k, W_m, W_k)$, where V denotes the nodes of entities; L_m denotes the set of common-entity edges and L_k denotes the set of world-knowledge edges; and W_m and W_k are the two sets of weights associated with L_m and L_k respectively.

Following G&S, for each common-entity edge $l_m \in L_m$, which connects two mentions of the same entity e appearing in different sentences s_x and s_y, we compute its weight as $w(e, s_x) \times w(e, s_y)$, where the value of $w(e, s_x)$ is based on the grammatical role of the entity e in the sentence s_x as follows: 3 for subject (S), 2 for object (O), 1 for other (X), and 0 for nothing($-$). When multiple mentions of the same entity appear with different grammatical roles in the same sentence, the role with the highest weight is chosen to represent the entity. For each world-knowledge edge $l_k \in L_k$, which connects two different entities e_{ix} and e_{jy} appearing in sentence s_x and s_y respectively, we consider three factors when assigning the weight to the edge: (1) semantic relatedness between e_{ix} and e_{jy}: higher relatedness leads to a higher weight; (2) the grammatical roles of e_{ix} and e_{jy} in s_x and s_y: different roles correspond to different weights; and (3) textual distance between e_{ix} and e_{jy}: longer distance results in a lower weight. Therefore we compute the weight of l_k between e_{ix} and e_{jy} as below:

$$w_k(e_{ix}, e_{jy}) = \frac{r(e_i, e_j) \times w(e_{ix}, s_x) \times w(e_{jy}, s_y)}{d(e_{ix}, e_{jy}) \times 2}, \quad (2)$$

where $w(e_{ix}, s_x)$ is associated to the grammatical role of e_{ix} in s_x as illustrated before; $d(e_{ix}, e_{jy})$ is the distance between e_{ix} and e_{jy}; and $r(e_i, e_j)$ is the semantic relatedness between e_{ix} and e_{jy} as shown in For-

mula 3 below. Note that the value of $r(e_i, e_j)$ is independent of the sentence in which e_i and e_j appear, so we denote it as $r(e_i, e_j)$ instead of $r(e_{ix}, e_{jy})$.

$$r(e_i, e_j) = \begin{cases} \dfrac{\log n(e_i, e_j)}{\max\limits_{l_{mn} \in L_k} \log n(e_m, e_n)} & \text{if } n(e_i, e_j) > 2, \\ 0 & \text{otherwise.} \end{cases} \tag{3}$$

where $n(e_i, e_j)$ corresponds to the number of world knowledge instances relating e_i and e_j. For instance, Figure 1a contains an edge between ***Gates*** (e_1 in s_1) and ***Microsoft*** (e_2 in s_3), in which $w(e_{11}, s_1)$ and $w(e_{23}, s_3)$ are 3, and $d(e_{11}, e_{23})$ is 2. Note that we consider the grammatical roles in both common edges and background knowledge edges because we treat them independently from each other. The grammatical information is important to both of these two kinds of edges.

4.1.2 Sentence graph

Figure 2d shows the sentence graph converted from the entity graph in Figure 2c. In our work, in order to incorporate world knowledge, we adopt an enriched representation of sentence graph, $G' = (V', L'_m, L'_k, W'_m, W'_k)$, where V' is the sentence nodes; L'_m denotes the set of common-entity edges (solid lines); L'_k denotes the set of world-knowledge edges (dashed lines), and W'_m and W'_k correspond to the weights associated with the edges in L'_m and L'_k.

Intuitively, the semantic relatedness between two sentences can be measured as the total relatedness of each entity pair in the two sentences. Therefore, in our enhanced sentence graph representation, for a pair of sentences s_x and s_y, the weight of their common-entity edge, $w'_m(s_x, s_y)$, is computed as $w'_m(s_x, s_y) = \sum_{e_i \in V_{xy}} w_m(e_{ix}, e_{iy})$, where V_{xy} is the set of entities shared by two sentences s_x and s_y, and $w_m(e_{ix}, e_{iy})$ is the weight of the corresponding common-entity edge in the entity graph (see Section 4.1.1). Similarly, the weight of their world-knowledge edge, $w'_k(s_x, s_y)$, is computed as $w'_k(s_x, s_y) = \sum_{e_i \in V_x, e_j \in V_y} w_k(e_{ix}, e_{jy})$, where V_x and V_y denote the set of entities in s_x and s_y respectively, and $w_k(e_{ix}, e_{jy})$ is the weight of the corresponding world-knowledge edge in the entity graph.

Algorithm 1: Reachability score computation.

Input: $G' = (V', L'_m, L'_k, W'_m, W'_k)$.
Output: The final reachability score S.

$n \leftarrow |V'|$
for $j = 1 \rightarrow n$ **do**
  $score(v'_j) \leftarrow 0$
for $j = 1 \rightarrow n$ **do**
  for $i = 0 \rightarrow j - 1$ **do**
    if $l'_m(i, j) \in L'_m$ **then**
      $score(v'_j) \leftarrow score(v'_i) + w'_m(i, j)$
    if $l'_k(i, j) \in L'_k$ **then**
      $score(v'_j) \leftarrow score(v'_i) + w'_k(i, j)$
return $S = \dfrac{\sum_{v'_j \in V' \wedge out(v'_j)=0} score(v'_j)}{|\{v'_j : v'_j \in V' \wedge out(v'_j) = 0\}|}$

4.1.3 Reachability score computation

Based on our sentence graph representation, we compute a reachability score for each sentence node. To produce a final coherence score for the text, we compute the ***average reachability score*** among those nodes whose out-degree is equal to 0 in the graph, rather than among all nodes, because of the intuition that, if a sentence node has no subsequent nodes, their reachability score therefore reflects the tightness between this sentence and the preceding part of the text. For a certain sentence node v'_j, its reachability score is defined as the sum of edge weights on all paths from the starting node (i.e., the first sentence) to v'_j, and the contribution of each path to the final reachability score depends on the total weight of that path as shown in Equation 4.

$$score(v'_j) = \sum_{v'_i \in V', v'_i \neq v'_j} (score(v'_i) + w'_m(i, j) + w'_k(i, j)) \tag{4}$$

where $score(v'_i)$ denotes the reachability score of v'_i, and $w'_m(i, j)$ and $w'_k(i, j)$ are the weights of the common-entity edge and the world-knowledge edge between v'_i and v'_j; if there is no such edge in the graph, the corresponding weight is set to 0.

Algorithm 1 summarizes our reachability score computation, in which the reachability score of each node is initially 0, and iteratively updated.

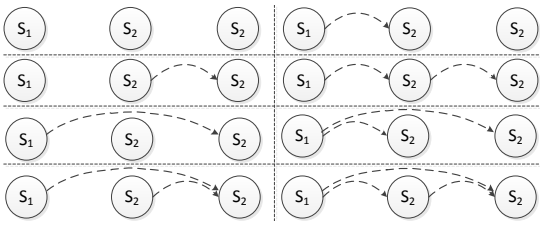


Figure 3: Eight patterns of how world knowledge is distributed among three adjacent sentences.

4.2 Supervised entity-based framework

As mentioned previously, numerous extensions have been proposed to the original entity-based model of B&L. However, those extensions mostly rely on entity matching and thus fail to incorporate the information from semantically related yet distinct entities. We propose a novel extension by introducing world knowledge to capture entity-wise relatedness.

Inspired by the original entity-based model, in which local coherence is reflected by the patterns of how entities act grammatically from one sentence to the next, we believe that local coherence can also be characterized by the patterns of how world knowledge relates a pair of sentences. Specifically, given a set of sentences, there are different patterns of how knowledge instances are distributed among them. We consider modeling those patterns within a window of 3 sentences, in which there are $2^3 = 8$ different distribution patterns, as shown in Figure 3. We then use the frequencies of these distribution patterns over the entire document as additional features into the entity-based model. In particular, for each particular distribution pattern b_k, its corresponding frequency $p(b_k) = \frac{|b_k|}{|V'| - 2}$, where $|b_k|$ is the number of occurrences of b_k in the sentence graph.

In addition to $p(b_k)$, we also compute another feature, $p(E)$, which is the frequency that two nodes are connected by certain world knowledge over the sentence graph, reflecting the overall semantic relatedness within the graph. $p(E)$ is computed as $p(E) = \frac{|L'_k|}{|V'|}$. With these world knowledge features incorporated into the original entity-based model, we obtain an enhanced model with an emphasis on semantic relatedness between different entities.

5 Experiments

To evaluate the impact of incorporating semantic relatedness, we conduct experiments on two datasets, each of which resembles a real sub-task in modeling text coherence: **sentence ordering** and **summary coherence rating**. Since text coherence is a relative concept rather than a binary distinction, in both tasks, we formulate the problem as pairwise preference ranking. Specifically, given a set of texts with different degrees of coherence, we train a ranker to prefer the more coherent text over the less coherent one. Performance is therefore measured as the fraction of correct pairwise rankings as recognized by the ranker. We use SVM^{light} (Joachims, 2002) with the ranking configuration to train and evaluate our models, with all parameters set to default values.

On both tasks, across two frameworks, supervised and unsupervised, we directly compare our modified model against the original one, i.e., B&L in the supervised framework and G&S in the unsupervised framework. In our experiments, we use the Stanford parser (Marneffe et al., 2006) to automatically extract the grammatical role for each entity mention.

5.1 Sentence ordering

The task of sentence ordering attempts to simulate the situation where, given a predefined set of information-bearing items, we need to determine the best order to arrange those items.

In this paper, we follow G&S and introduce CoNLL 2012[2] (Pradhan et al., 2012) as our dataset, which is composed of documents from multiple news sources. For each text, we randomly shuffle its sentences to generate 20 permutations with incorrect sentence order. For a fair comparison, we also evaluate our model on a filtered subset of documents with an average length of 31.8 sentences. Therefore, our dataset contains 72 documents and $72 \times 20 = 1440$ permutations, among which the shortest one contains 25 sentences. For our enhanced graph-based model (introduced in Section 4.1), which is purely unsupervised, we evaluate our model over the entire dataset. For our enhanced entity-based model (introduced in Section 4.2), which is purely supervised, we use half of the complete CoNLL dataset for training (237 documents plus permutations) and use half

[2]http://conll.cemantix.org/2012/data.html

of the filtered subset (36 documents plus permutations). The training and test sets do not overlap.

In this task, each training and test instance is composed of a pair of a source document and one of its permutations, and the source document is always considered more coherent than its permutation.

5.2 Summary coherence rating

The second task is summary coherence rating, in which, given a pair of summaries about the same set of source documents, we determine the ranking of these two summaries based on their degrees of coherence. The performance of the model is assessed by comparing model-induced rankings against the rankings given by human judges. We use the same dataset (DUC 2003) as B&L and G&S did, which consists of summaries generated either by human writers or by automatic summarization systems. Each summary was given a coherence score by averaging among seven judges. Often, machine-generated summaries receive low coherence scores because they contain sentences taken out of context and thus display problems with respect to coherence.

This dataset consists of 16 input document clusters, each of which is associated with five machine-generated summaries along with a human-written summary. In total, we have 96 summaries (for more details, see B&L). We form pairwise rankings by taking any two summaries originating from the same document cluster, given that the two summaries receive different coherence scores: 144 of the resulting rankings are used for training and 80 are for testing.

5.3 Experiment results

In this section, we demonstrate the performance of our models with world knowledge encoded in one of the two ways: paths in a sentence graph or features in an entity grid. We compare our models against the original graph-based model (G&S) and entity-based model (B&L). The evaluation is conducted on the two tasks, sentence ordering and summary coherence rating, and the accuracy is the fraction of correct pairwise rankings.

Table 1 shows the performance of various models on both tasks. The first section shows the results of G&S's graph-based local coherence model, including the performance reported in their original paper and that achieved by our re-implementation, representing the effect with no world knowledge encoded. The second section shows the performance of our two graph-based models with world knowledge encoded. *Graph model + K* is the basic model with world knowledge encoded, but coherence is simply measured as the average out-degree as in G&S's approach. *Graph model + K + Avg_R* replaces the out-degree measurement by our *average reachability score* (described in Section 4.1.3), which measures coherence in a more sophisticated way. The third section shows the results of B&L's entity-based local coherence model, including the originally reported performance and that obtained by our re-implementation, in which no world knowledge features are included. The last section, *Entity model + K*, shows the result of entity-based model with our world knowledge features encoded. Note that the random baseline of both tasks is 50%.

Model	**SO**	**SCR**
Graph model (G&S)	88.9	80.0
Graph model (Implemented)	89.6	48.8
Graph model + K	91.3**	50.0
Graph model + K + Avg_R	**93.4****	**55.0***
Entity model (B&L)	88.9	83.8
Entity model (Implemented)	93.7	90.0
Entity model + K	**95.1****	**91.3**

Table 1: Accuracies (%) of various models on the two tasks, sentence ordering (*SO*) and summary coherence rating (*SCR*). Models that perform significantly better than their corresponding re-implemented basic models are denoted by ** ($p < .01$) or * ($p < .05$), verified using paired *t*-test.

Firstly, for graph-based models, our *Graph model + K* outperforms the original models, suggesting that world knowledge is truly helpful for capturing more coherence information[3]. Moreover, by intro-

[3]The large discrepancy between the performance reported by G&S and that of our re-implementation in *Task 2* is due to the fact that G&S experimented with a set of specially formed summary pairs (see their paper for detail), which we have no access to. They also did not give sufficient details about how they constructed those summary pairs, which has a great impact on the final result. This made it difficult for us to fully re-implement their experiment. So we use B&L's set of summary pairs, which are generated randomly and are more difficult to distinguish, which explains our differing results from theirs.

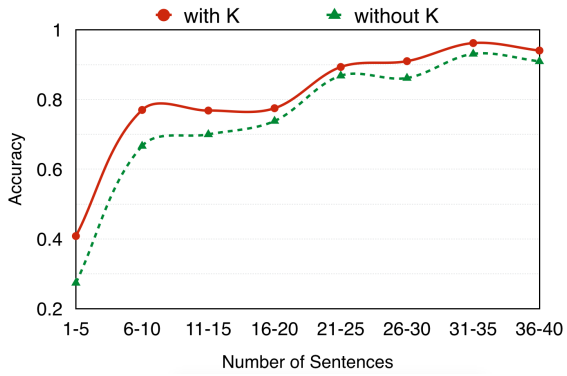


Figure 4: Graph-based models, with and without world knowledge (labeled as *With K* and *Without K*), tested on sets with different numbers of sentences.

ducing the scoring scheme of *average reachability score*, our *Graph model + K + Avg_R* achieves the best performance among all graph-based models.

Secondly, for entity-based models, our enhanced model with knowledge features encoded also achieves superior performance than our re-implemented model, again confirming the usefulness of world knowledge. Interestingly, we observe that our re-implementation obtains higher accuracy compared to the performance reported by B&L. This is partly due to the fact that the documents in our dataset have an average length of 31.5 sentences, which are longer than those used in B&L's experiments. We will further discuss this problem in Section 5.4 and show that document length is an important factor to the overall performance.

However, on the task of summary coherence rating, the difference between our extended models and the original ones is generally not significant, primarily due to the fact that the sample size for this task is too small, i.e., 80 pairwise rankings.

5.4 Effect of document length

5.4.1 Effect of document length on the overall performance

We further analyze the impact of document length on the task of sentence ordering. We partition our original dataset, which consists of 214 documents and their permutations, into 8 non-overlapping subsets, according to the length of documents: 1-5, 6-10,..., and 36-40 sentences. To illustrate the correlation between the performance and the length of document, we test our models with and without world knowledge encoded on each subset separately. Since the size of the available training data in each subset is relatively small, the supervised entity-based model suffers from sparsity. Therefore, we focus on the unsupervised graph-based models only.

Model	Accuracy (%)
Graph Model	27.4
Graph Model + K	**44.2**
Entity Model	65.8
Entity Model + K	**71.1**

Table 2: Performance of various models with and without world knowledge in the sentence ordering task, tested on short documents with 1–5 sentences.

Figure 4 shows the performance on different subsets. We can see that the performance of both models generally improves as the number of sentences increases. This observation is quite intuitive, because the longer a document is, the higher the chance is that, after being shuffled, adjacent sentences in the resulting permutation would be completely irrelevant to each other. Therefore, for longer documents, it is much easier for the model to distinguish a permutation from its source document.

5.4.2 Effect of document length on the model with world knowledge

Moreover, we also observe that the document length has a non-universal effect, in terms of how the model could benefit from incorporating world knowledge. Specifically, we find that world knowledge has a greater effect on short documents, as demonstrated in Table 2. Evaluated on a set of documents composed of 30 extremely short documents only (1–5 sentences), we see that our enhanced graph-based model is able to improve the performance by 16.8% over the basic model, and our enhanced entity-based model achieves 5.3% improvement (both differences are significant at $p < .01$). We postulate that it is primarily because a document with fewer sentences tends to shift to another subtopic immediately without elaborating on the previous one, and strict entity matching would find it difficult to establish coherent transitions between them. Therefore, the contribution from semantic relatedness tends to dominate the overall performance.

6 Conclusions and future work

In this paper, for the evaluation of text coherence, we go beyond strict entity matching and model the semantic relatedness between distinct entities through the use of world knowledge. Specifically, we incorporate world knowledge into two existing frameworks: (1) the unsupervised graph-based model (G&S), and (2) the supervised entity-grid model (B&L). Across the two frameworks, on both of our evaluation tasks, sentence ordering and summary coherence rating, our enhanced models with world knowledge encoded are shown to be stronger than the corresponding basic models, confirming that semantic relatedness is truly important for coherence modeling and such relatedness can be effectively captured by world knowledge. Moreover, we observe that world knowledge is particularly useful for short documents in sentence ordering, as it provides additional clues to relate sub-topics in the text.

In our future work, we wish to explore the effect of our world knowledge in conjunction with discourse relations. Specifically, we plan to incorporate world knowledge into the framework of discourse role matrix (Lin et al., 2011; Feng et al., 2014). In addition, we also plan to develop a more sophisticated feature encoding by distinguishing different types of predicates in world knowledge triples.

Acknowledgments

We would like to thank Mao Zheng and Yanyan Zhao for their great help. This work was partly supported by National Natural Science Foundation of China via grant 61133012, the National 863 Leading Technology Research Project via grant 2012AA011102 and the National Natural Science Foundation of China Surface Project via grant 61273321.

References

Regina Barzilay and Mirella Lapata. 2008. Modeling local coherence: An entity-based approach. *Computational Linguistics*, 34(1):1–34.

Regina Barzilay, Noemie Elhadad, and Kathleen R. McKeown. 2002. Inferring strategies for sentence ordering in multidocument news summarization. *Journal of Artificial Intelligence Research*, 17(1):35–55.

Kurt Bollacker, Colin Evans, Praveen Paritosh, Tim Sturge, and Jamie Taylor. 2008. Freebase: a collaboratively created graph database for structuring human knowledge. In *Proceedings of the 2008 ACM SIGMOD international conference on Management of data*, pages 1247–1250. ACM.

Ludovic Denoyer and Patrick Gallinari. 2007. The Wikipedia XML corpus. In *Comparative Evaluation of XML Information Retrieval Systems*, pages 12–19. Springer.

Micha Elsner and Eugene Charniak. 2008. Coreference-inspired coherence modeling. In *Proceedings of the 46th Annual Meeting of the Association for Computational Linguistics on Human Language Technologies: Short Papers*, pages 41–44. Association for Computational Linguistics.

Micha Elsner and Eugene Charniak. 2011. Extending the entity grid with entity-specific features. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies: short papers-Volume 2*, pages 125–129. Association for Computational Linguistics.

Anthony Fader, Stephen Soderland, and Oren Etzioni. 2011. Identifying relations for open information extraction. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, pages 1535–1545. Association for Computational Linguistics.

Vanessa Wei Feng, Ziheng Lin, and Graeme Hirst. 2014. The impact of deep hierarchical discourse structures in the evaluation of text coherence. In *Proceedings of the 25th International Conference on Computational Linguistics*, pages 940–949.

Katja Filippova and Michael Strube. 2007. Extending the entity-grid coherence model to semantically related entities. In *Proceedings of the Eleventh European Workshop on Natural Language Generation*, pages 139–142. Association for Computational Linguistics.

Camille Guinaudeau and Michael Strube. 2013. Graph-based local coherence modeling. In *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics*, pages 93–103.

Johannes Hoffart, Fabian M Suchanek, Klaus Berberich, and Gerhard Weikum. 2013. Yago2: a spatially and temporally enhanced knowledge base from Wikipedia. *Artificial Intelligence*, 194:28–61.

Thorsten Joachims. 2002. Optimizing search engines using clickthrough data. In *Proceedings of the Eighth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 133–142. ACM.

Ziheng Lin, Hwee Tou Ng, and Min-Yen Kan. 2011. Automatically evaluating text coherence using discourse

relations. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies-Volume 1*, pages 997–1006. Association for Computational Linguistics.

M. Marneffe, B. Maccartney, and C. Manning. 2006. Generating typed dependency parses from phrase structure parses. In *Proceedings of the 5th International Conference on Language Resources and Evaluation*, volume 6, pages 449–454. European Language Resources Association (ELRA).

Neil McIntyre and Mirella Lapata. 2010. Plot induction and evolutionary search for story generation. In *Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics*, pages 1562–1572. Association for Computational Linguistics.

Eleni Miltsakaki and Karen Kukich. 2004. Evaluation of text coherence for electronic essay scoring systems. *Natural Language Engineering*, 10(1):25–55.

Sameer Pradhan, Alessandro Moschitti, Nianwen Xue, Olga Uryupina, and Yuchen Zhang. 2012. CoNLL-2012 shared task: Modeling multilingual unrestricted coreference in ontonotes. In *Joint Conference on EMNLP and CoNLL-Shared Task*, pages 1–40. Association for Computational Linguistics.

Muyu Zhang, Bing Qin, Ting Liu, and Mao Zheng. 2014. Triple based background knowledge ranking for document enrichment. In *Proceedings of the 25th International Conference on Computational Linguistics*, pages 917–927.