Using a Graph-based Coherence Model in Document-Level Machine Translation

Leo Born[†], **Mohsen Mesgar**[‡] and **Michael Strube**[‡]
† Department of Computational Linguistics, Heidelberg University
Heidelberg, Germany
born@cl.uni-heidelberg.de

‡ Heidelberg Institute for Theoretical Studies gGmbH
Heidelberg, Germany
(mohsen.mesgar|michael.strube)@h-its.org

Abstract

Although coherence is an important aspect of any text generation system, it has received little attention in the context of machine translation (MT) so far. We hypothesize that the quality of document-level translation can be improved if MT models take into account the semantic relations among sentences during translation. We integrate the graph-based coherence model proposed by Mesgar and Strube (2016) with Docent[1] (Hardmeier et al., 2012; Hardmeier, 2014) a document-level machine translation system. The application of this graph-based coherence modeling approach is novel in the context of machine translation. We evaluate the coherence model and its effects on the quality of the machine translation. The result of our experiments shows that our coherence model slightly improves the quality of translation in terms of the average Meteor score.

1 Introduction

Coherence represents semantic connectivity of texts with regard to grammatical and lexical relations between sentences. It is an essential part of natural texts and important in establishing structure and meaning of documents as a whole.

It is crucial for any text generation system to generate coherent texts. For instance in real machine translation systems, we desire to translate a document, which consists of several sentences, from a source language to a target language. Current machine translation systems (as an instance of text generation systems) mostly focus on the sentence-level translation. Indeed, the state-of-the-art machine translation models perform well on sentence-level translation (Bahdanau et al., 2015; Sennrich et al., 2017). However, it is insufficient to just sequentially and independently translate sentences of the source document and concatenate them as the translated version. The translated sentences should be coherently connected to each other in the target document as well.

From a linguistic point of view also the discourse-wide context must be taken into account to have a high-quality translation (Hatim and Mason, 1990; Hardmeier et al., 2012). The current paradigm of machine translation needs to be improved as it does not consider any discourse coherence phenomena that establish a text's connectedness (Sim Smith et al., 2015).

One of the active research topics in modeling coherence focuses on entity connections over sentences based on Centering Theory (Grosz et al., 1995). Previous research on coherence modeling shows its application mainly in readability assessment (Barzilay and Lapata, 2008; Pitler and Nenkova, 2008). Recently, Parveen et al. (2016) showed that the graph-based coherence model can be utilized to generate more coherent summaries of scientific articles.

The main goal of this paper is to integrate coherence features with a statistical machine translation system to improve the quality of the output translation. To achieve this goal, we combine the graph-based coherence representation by Guinaudeau and Strube (2013) and its extensions (Mesgar and Strube, 2015, 2016) into the document-level machine translation decoder *Docent* (Hardmeier et al., 2012, 2013).

Docent defines an initial translation of the source document and modifies the translation of sentences aiming to maximize an objective function. This function measures the quality of the

[1] https://github.com/chardmeier/docent

S1: But the noise didn't disappear.

S2: The mysterious noise that Penzias and Wilson were listening to turned out to be the oldest and most significant sound that anyone had ever heard.

S3: It was cosmic radiation left over from the very birth of the universe.

S4: This was the first experimental evidence that the Big Bang existed and the universe was born at a precise moment some 14.7 billion years ago.

S5: So our story ends at the beginning – the beginning of all things, the Big Bang.

Table 1: Excerpt of a TED talk (ID: 1177) from the DiscoMT 2015 training data.

translated document after each modification. We propose to update the objective function of Docent such that it takes into account the coherence of the translated document too. We quantify the coherence level of the translated document using graph-based coherence features. We show that integrating coherence features improves the quality of the translation in terms of the Meteor score.

We start with the relevant background literature (Section 2). We then describe the graph-based coherence model and how we integrate its coherence features with Docent (Section 3). Section 4 outlines the datasets and the experimental setup. We discuss results in Section 5. Conclusions and possible future work are in Section 6.

2 Related Work

2.1 Entity Graph

Guinaudeau and Strube (2013) present a graph-based version of the entity grid (Barzilay and Lapata, 2008). It models the interaction between entities and sentences as a bipartite graph. In this representation, one set of nodes corresponds to sentences, whereas the other set of nodes corresponds to entities in a document. Table 1 shows a sample text from our training data and Figure 1 the bipartite entity-graph representation of it.

Coherence is measured over the one-mode projection on sentence nodes. The one-mode projection is the graph in which the sentence nodes are connected to each other if and only if they have at least one entity in common (see Figure 2). The coherence of a text T can then be measured by computing the average outdegree of the projection graph. Outdegree of a node is the number of edges that leave the node. The average outdegree is the sum of outdegree of all nodes in the one-mode projection graph divided by the number of sentences.

Mesgar and Strube (2015) evaluate this model for readability assessment. They show that the average outdegree is not the best choice for quantifying the coherence. They propose to encode coherence as the connectivity structure of sentence nodes in a projection graph. So they represent the connections among sentences of each document in the corpus with its projection graph; then they mine all possible subgraphs of these graphs. These subgraphs resemble what the linguistic literature terms *thematic progression* (Daneš, 1974) as subgraphs represent connections between sentences following a certain pattern. Mesgar and Strube (2015) call these subgraphs *coherence patterns*. The connectivity structure of a projection graph can be modeled by the frequency of subgraphs in each graph. These frequencies are called *coherence features*. Mesgar and Strube (2015) show that these coherence features, obtained from frequency of subgraphs of projection graphs of the entity graphs, can assess readability better. Figure 3 illustrates four possible subgraphs with three nodes. The pool of possible subgraphs can be expanded to encompass any arbitrary number of nodes, so-called k-node subgraphs.

Mesgar and Strube (2016) extend the entity graph to the lexical graph: two sentences may be semantically connected because at least two words of them are semantically associated to each other. They compute semantic relatedness between all content word pairs using *GloVe* word embeddings (Pennington et al., 2014). If there is a word pair whose word vectors have a cosine relatedness greater than a threshold, two sentences are considered to be connected. They quantify the coherence of texts via frequency of subgraphs of the lexical graphs. It outperforms the entity graph coherence model on readability assessment.

Parveen et al. (2016) show that coherence patterns can be mined from a corpus and those can get weighted based on their frequencies in the corpus. They use the extracted coherence patterns and their weights to generate a coherent summary from scientific documents. Using a human evaluation, they show that coherence patterns are more powerful than average outdegree to encode coherence for automatic summarization.

Here we check if these coherence features (i.e., average outdegree and frequency of coherence patterns) of graph-based models can assist

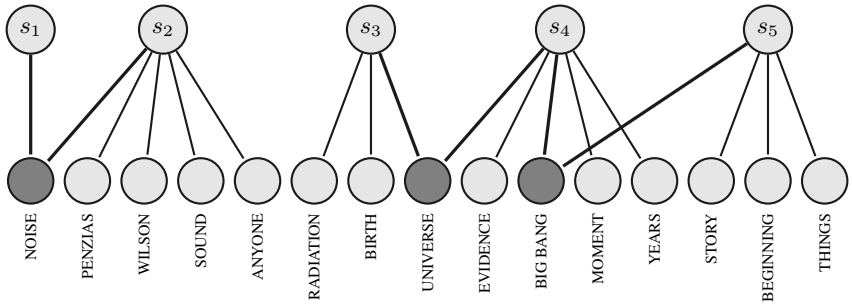

Figure 1: The entity graph representation of the text in Table 1. Dark entities are shared by the sentences.

Figure 2: Unweighted projection graph of the entity graph in Figure 1. The nodes are connected based on whether sentences share an entity or not, whereas the edge direction follows sentence order.

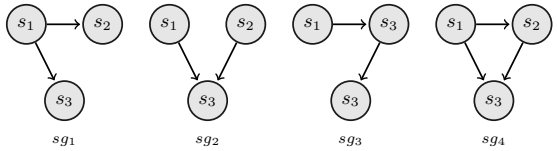

Figure 3: All possible directed 3-node subgraphs. The edge directions indicate the order of sentences in the text.

document-level machine translation as another, and more difficult, text generation system. We can also evaluate which feature is more beneficial for machine translation.

2.2 Coherence in Machine Translation

Coherence modeling in machine translation is an (almost) desideratum . To the best of our knowledge, there are only a handful of publications in this direction. The one relevant to our approach is the work by Lin et al. (2015) as it constitutes an application of a coherence model in the context of machine translation, as opposed to more theoretical papers on the state of coherence in machine translation (Sim Smith et al., 2016).

Lin et al. (2015) develop a sentence-level Recurrent Neural Network Language Model (RNNLM) that takes a sentence as input and tries to predict the next one based on the sentence history vector. By modeling sequences of sentences, the vector is able to model local coherence within RNNLM.[2] Given the 10-best results of all sentences from the decoder, their system then selects the best translation for the first sentence. Given that translation, they score all translation candidates of the second sentence based on coherence and select the best one. They repeat this for all sentences in the document.

This approach, however, can be considered linguistically weak as it only measures coherence after the translation and does not consider it as a part of the text generation process. As coherence, however, is a fundamental need for any text generation system (Barzilay and Lapata, 2008), this motivates us to go beyond a simple re-ranking approach and integrate the coherence measure directly into the decoding process of machine translation.

3 Method

3.1 Docent

We use *Docent* (Hardmeier et al., 2012, 2013) as the baseline. It explicitly has no notion of coherence. Docent is a document-level decoder that treats a translation not as a bag of sentences but instead has a translation hypothesis for the whole document at each step. The initial hypothesis can either be generated randomly from the translation table or it can be initialized with the result of any standard sentence-level decoder such as Moses (Koehn et al., 2007).

Docent first independently translates all sentences of the input document. Then it starts to modify the translation of sentences with respect to the other translated sentences. Three basic operations modify the translation of sentences: *change-phrase-translation*, *swap-phrases*, and *resegment*. *Change-phrase-translations* replaces the translation of a single phrase with a random translation for the same source phrase. *Swap-phrases* changes the word order without affecting the phrase translations by exchanging two phrases in a sentence.

[2] They consider the "log probability of a given document as its coherence score" (Lin et al., 2015).

The third operation, *resegment*, is able to generate from a number of phrases a new set of phrases covering the same span. Docent checks the quality of the modified translation by an objective function that takes the modified translation of the document (the so-called state of the translated document) as its input and maps it to a real number. If the value of the objective function increases then Docent accepts the applied operation.

The main advantage of Docent is that the objective function can be defined over the whole document (Hardmeier et al., 2012). This allows us to integrate our new document-level coherence features with Docent. More formally, the overall document state S is modeled as a sequence of sentence states:

$$S = S_1 S_2 ... S_N, \quad (1)$$

where N is the number of sentences and S_i is the translation (hypothesis) of the i^{th} source sentence. A scoring function $f(S)$ maps a state to a real number. The scoring function can be further decomposed into a linear combination of K feature functions $h_k(S)$, each with a constant weight λ_k, such that

$$f(S) = \sum_{k=1}^{K} \lambda_k h_k(S). \quad (2)$$

Docent uses *simulated annealing*, a stochastic variant of the hill climbing algorithm (Khachaturyan et al., 1981), for either accepting or rejecting operations for maximizing its objective function (Hardmeier, 2012) .

Docent already implements some sentence-local feature models that are similar to those found in traditional sentence-level decoders. These include phrase translation scores provided by the phrase table (Koehn et al., 2003), n-gram language model scores implemented with *KenLM* (Heafield, 2011), a word penalty score, and an unlexicalised distortion cost model with geometric decay (Koehn et al., 2003).

Our idea is to add a new document-level coherence function $h_{coh}(S)$, namely a graph-based coherence model to the objective function represented in Equation. 2. In the next subsection, we describe this model in more detail.

3.2 Graph-based Coherence Model

Our coherence model is based on the lexical graph representation (Mesgar and Strube, 2016). For any given document, we first filter out stop words using the provided stop word list by Salton (1971). Then, we calculate the cosine relatedness of all remaining word pairs of all sentence pairs using the 840 billion token pre-trained word embeddings of *GloVe* (Pennington et al., 2014). For every out-of-vocabulary word, we assign a random 300-dimensional vector that is memorized for its next occurrence. Based on this, we represent the lexical relations among sentences via graphs. If at least two words in the sentences are related, we choose the relation between those two words whose embeddings have the maximum cosine value. In order to make the graph not too dense, we filter out those edges whose strengths are below a certain threshold.

However, in contrast to Mesgar and Strube (2016), we use a different threshold for graph construction. They use a threshold of 0.9, but we find this too strict on allowing the graph structure to change in the direction of more coherent texts. We choose a lower threshold, 0.85, to let the model consider more connections and more lexical variations (i.e., synonyms) in the translation.

We encode coherence by frequency of coherence patterns in these graphs.

3.3 Integrating the Coherence Model With Docent

For extracting coherence patterns we use the target documents[3] of the training set of the DiscoMT dataset. We extract all k-node subgraphs for $k \in \{3, 4, 5\}$. We limit the size of subgraphs to 3-, 4-, and 5-node as Mesgar and Strube (2016) report declining results for subgraphs with $k > 5$.

We also calculate a respective weight for each pattern from lexical graph representations of DiscoMT training target documents.

We base our coherence patterns on the characteristics of the target language as there is a theory within Translation Studies that "textual relations obtaining in the original are often modified [...] in favour of (more) habitual options offered by a target culture" (Toury, 1995). Toury (1995) calls this the *law of growing standardization* which seeks to describe and explain the acceptability of the translation in the receiving culture (Venuti, 2004). This law seems suitable in the context of subgraph mining as it is also already reflected in the language model of any MT system (Lembersky et al., 2012).

For computing the weights of subgraphs, we divide the count of each k-node subgraph by the to-

[3] We experiment on translation from French to English.

tal counts of subgraphs for that k. For each k, this gives the following vector:

$$\varphi(sg^k, G) = (w(sg^k_1, G), ..., w(sg^k_m, G)), \quad (3)$$

where formally

$$w(sg^k_i, G) = \frac{count(sg^k_i, G)}{\sum_{sg^k_j \in (sg^k_1, ..., sg^k_m)} count(sg^k_j, G)}. \quad (4)$$

These weights are then used as weights of coherence features in the coherence function, $h_{coh}(S)$, that quantifies the connectivity structure of sentences of an intermediate state of the translated document in Docent during evaluation on the test set of DiscoMT.

So, given the coherence graph representation of an intermediate state of the translated document (during the test phase), G_S, and the set of all extracted subgraphs of the training documents, $FSG = \{sg^k_1, sg^k_2, ..., sg^k_m\}$ where $k \in \{3, 4, 5\}$, and their weights, $h_{coh}(S)$ is defined as follow:

$$h_{coh}(S) = \sum_{sg^k_i \in FSG} count(sg^k_i, G_S) \cdot w(sg^k_i). \quad (5)$$

We use this score – which multiplies the frequency of each subgraph in each state (coherence feature) of the translated document with its weight according to its frequency in the training documents and sums this up for all subgraphs – as our feature model score of our coherence model.

4 Experiments

4.1 Datasets

We use the WMT 2015 (Bojar et al., 2015) dataset for training and development of the sentence-level translation and language models[4], and the DiscoMT 2015 Shared Task (Hardmeier et al., 2015) dataset for mining subgraphs (coherence patterns) and as our test data (Table 2). We run experiments on the language pair French-English. Coherence patterns are extracted from the 1551 DiscoMT *training* documents using *GloVe* word embeddings. We extract all k-node subgraphs for $k \in \{3, 4, 5\}$ using *GASTON*[5] (Nijssen and Kok, 2004, 2005).

[4]We use Moses to translate sentences independently and initialize the translation state in Docent.

[5]http://liacs.leidenuniv.nl/~nijssensgr/gaston/iccs.html.

We use the twelve test documents of DiscoMT as the test data because these are much longer, on the document level, than the WMT test data. The average number of sentences of the WMT test data is 20, whereas for DiscoMT it is 174 sentences. Thus it is a more difficult test set for our experiments.

	train	**dev**	**test**
# of docs	-	-	12
# of sent.	200,239	3,003	2,093
avg. # of sent. per doc	-	-	174
# of tokens	4,458,256	63,778	48,122

Table 2: Statistics on the datasets used. *train* is the news commentary v10 corpus, *dev* is the 2012 newstest development data, and *test* is the DiscoMT 2015 test data. The number (#) of tokens corresponds to the English (target) side.

4.2 Experimental Setup

We train our systems using the *Moses* decoder (Koehn et al., 2007). After standard preprocessing of the data, we train a 3-gram language model using *KenLM* (Heafield, 2011). We use the *MGIZA++* (Gao and Vogel, 2008) word aligner and employ standard *grow-diag-fast-and* symmetrization. Tuning is done on the development data via *minimum error rate training* (Och, 2003).

After training the language model and creating the phrase table with Moses, we use these to initialize our translation systems. We use the *lcurve-docent* binary of Docent, which outputs Docent's learning curve, i.e., files for the intermediate decoding states. This additionally allows us to investigate the learning curves with regard to how our coherence feature behaves over time.

We prune the translation table by only retaining all phrase translations with a probability greater than 0.0001 during training. In our configuration file for Docent, we set to use the simulated annealing algorithm with a maximum number of 16,384 steps[6] and the following features: *geometric distortion model*, *word penalty cost*, *OOV-penalty cost*, *phrase table*, and the *3-gram language model*.

[6]We choose this threshold to make a balance between processing time and translation performance.

4.3 Evaluation Metrics

We follow the standard machine translation procedure of evaluation, measuring *BLEU* (Papineni et al., 2002) for every system. BLEU is an *n*-gram based co-occurrence metric that operates with modified *n*-gram precision scores. The document *n*-gram precision scores are averaged using the geometric mean of these scores with *n*-grams up to length *N* and positive weights summing to one. The result is multiplied by an exponential *brevity penalty factor* that penalizes a translation if it does not match the reference translations in length, word choice, and word order.

We also calculate *Meteor* (Lavie et al., 2004; Denkowski and Lavie, 2014) as it is a widely used evaluation metric as well. In contrast to BLEU, Meteor is a word-based metric that takes recall into account as well. Meteor creates a word alignment between a pair of strings that is incrementally produced using a sequence of various word-mapping modules, including the *exact* module, the *Porter stem* module, and the *WordNet synonymy* module (Lavie and Agarwal, 2007).

Because Meteor has been shown to have a higher correlation with human judgements than BLEU (Lavie et al., 2004), it is a useful alternative evaluation metric for our purposes. As it also considers stemmed words and information from WordNet to determine synonymous words between a candidate and a reference translation, the metric is interesting with regard to surface variation with the same semantic content and how this affects the evaluation of our coherence model (as its graph construction is semantically grounded).

5 Results

5.1 Mined Coherence Patterns Analysis

We represent each English document of the training set of the DiscoMT dataset by a graph (as described in Section 3.2). As a result, instead of a set of documents we have a set of graphs. Then we extract all occurring subgraphs in these graphs as coherence patterns. We mine subgraphs with $3, 4, 5$ nodes.

All 3-node subgraphs exist in the graph representation of the training documents. It is because these subgraph are small and it is very likely that they occur in the graph representation of the large DiscoMT documents.

The mined 4-node subgraphs are shown in Figure 4. Although the frequency of these patterns

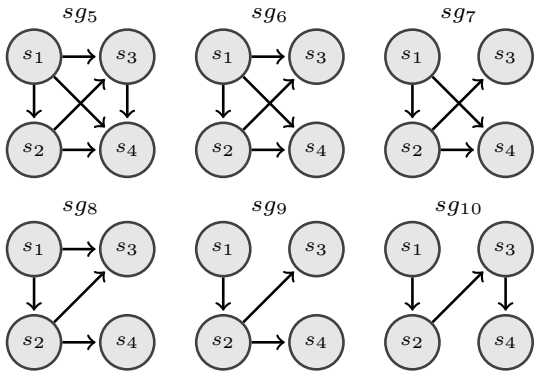

Figure 4: The mined 4-node subgraphs.

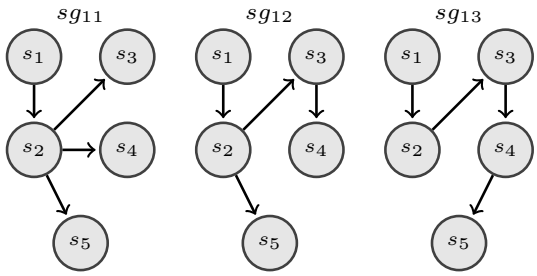

Figure 5: The mined 5-node subgraphs.

encode coherence in our model, the existence of these patterns can be linguistically interpreted too. For example, sg_{10} models the smooth shift in the topic of a sequence of sentences (Mesgar and Strube, 2015). The rest of the patterns have a common property: a sentence introduces some topic and the following sentences are about this topic. For instance, in sg_6, topics in the first sentence are developed by the rest of the sentences.

The mined 5-node subgraphs are shown in Figure 5. The expansion of a topic is much clearer here in sg_{11}. The subgraph sg_{13} is very similar to sg_{10} following the notion of the topic shift. This is somehow expected because the DiscoMT documents are obtained from TED talks. These talks are mostly given by professional speakers. They have to move smoothly from one topic to the next topic in a short sequence of sentences. This confirms the existence of the linear chain pattern in the 4-node and 5-node patterns.

We analyze the change of the frequencies of the subgraphs during the MT decoding phase. For example, on document 9 the subgraph sg_1 of the 3-node subgraphs occurs one more time in the CM model. It is worthwhile to note that the increase of the frequency of sg_1 is compatible with its positive correlation with readability scores of documents

Document ID	BLEU (BL)	BLEU (CM)	Meteor (BL)	Meteor (CM)
(#1) 1756	21.87	**21.93**	61.47	**61.52**
(#2) 1819	16.49	16.49	62.25	62.25
(#3) 1825	24.86	24.86	**66.34**	66.32
(#4) 1894	17.08	17.08	57.20	57.20
(#5) 1935	20.11	20.11	62.83	62.83
(#6) 1938	**20.43**	20.41	**63.53**	63.48
(#7) 1950	**23.27**	23.26	**63.48**	63.46
(#8) 1953	**20.78**	20.66	**61.65**	61.64
(#9) 1979	15.25	**15.26**	55.68	**55.69**
(#10) 2043	18.27	18.27	56.42	**56.47**
(#11) 2053	30.65	30.65	69.13	69.13
(#12) 205	13.79	13.79	52.68	52.68
Average	**20.24**	20.23	61.01	**61.06**

Table 3: Results of the coherence model (*CM*) compared to the baseline (*BL*) on the DiscoMT test set (highest values are marked in bold). The scores of the entity graph model using average outdegree as coherence feature are identical to the baseline model. The differences are not statistically significant ($p = 0.05$) using Student's *t*-test (Student, 1908).

in the readability assessment experiment done by Mesgar and Strube (2015). For the documents 1 and 10 the frequency of subgraphs are constant during decoding. It might be because the connectivity of sentences is already compatible with the training documents and our coherence features push the Docent model to reject operations that might disturb the structure. The decrease in the number of accepted operations for these two documents by the CM model (represented in Table 4) supports this.

5.2 Machine Translation Metrics Analysis

We evaluate the model on the test set of the DiscoMT dataset. As the baseline, we use the coherence-blind Docent and compare it against a system with the additional document-level coherence features.

First we try the entity graph model with the average outdegree as the coherence feature. The BLEU and Meteor scores of this model are identical to the baseline. This means that the average outdegree is not a good representative of coherence. That was also shown by Mesgar and Strube (2015) for the readability assessment task.

Next, we try the lexical graph representation of documents and frequency of coherence patterns as the coherence features.

The results of the baseline (BL) and our coherence model (CM) in terms of BLEU and Meteor scores are shown in Table 3.

Compared to the baseline, results for about half of the documents do not change in terms of BLEU. For two documents, the coherence model improves the BLEU score, whereas for three documents it diminishes. Overall, the average BLEU score of the coherence model is slightly lower than that of the baseline.

The Meteor score of the coherence model is better on three documents. The coherence model achieves the best overall result in terms of the averaged Meteor score. The coherence model does not improve the Meteor score on four documents.

We interpret these observations as follows: First, the coherence patterns can model the coherence property of texts better than average outdegree. This is compatible with the reported results by Mesgar and Strube (2015) and Parveen et al. (2016) that, respectively, show that coherence patterns are more informative for readability assessment and multi-document summarization. However, our results also indicate that they are not that powerful for a more difficult task like machine translation (Sim Smith et al., 2016).

Second, the obtained improvement of our coherence model, which is augmented with some document-level features, especially on the Meteor score confirms this hypothesis that the quality of the machine translation can be improved if the MT model is informed by the document-level context.

The third interpretation is about the validity of these traditional metrics that were constructed

in the context of sentence-level decoding. This means that these MT scores might not be that much appropriate to measure the global translation quality, especially with regard to discourse coherence. As a future work, we are going to do a human evaluation on this.

Table 4 indicates the number of accepted *change-phrase-translation* operations by Docent in a comparison between the baseline and the coherence model. For both models, the number of accepted operations is very close.

Document 1 is one of the documents where the coherence model outperforms the baseline and it is tempting to assume that the score difference stems from the one operation not accepted by the coherence model. Indeed, the only detectable difference in the two translations is in one sentence only (see its output translations in Table 5). The coherence features might prevent the translation model to change the translation of *thought for*, which is identical with the reference translation.

Similarly, for document 10 the CM model accepts one less operation than the baseline model and it, again, helps the model to obtain a higher Meteor score. Interestingly, the BLEU score on these two documents remains the same, so the score difference is likely a result of a more semantic change in translation. For the document 9 the CM model improves the MT scores by accepting more operations than the baseline model. For documents 3, 6 and 8 the accepted operations by the CM model reduce the MT scores.

Finally, supported operations in Docent seem insufficient to change the structure of graphs. From the three basic operations Docent uses, the two operations *swap-phrases* and *resegment* may not change the graph structure. *Change-phrase-translation*, however, has the potential to actually change the graph structure by either choosing an alternative translation of a word that is either not connected to any other words anymore or that conversely connects to another word within the text.

Document ID	**# of accepted operations**	
	BL	**CM**
(#1) 1756	22	21
(#2) 1819	18	18
(#3) 1825	22	21
(#4) 1894	25	25
(#5) 1935	21	21
(#6) 1938	30	33
(#7) 1950	59	59
(#8) 1953	29	32
(#9) 1979	25	26
(#10) 2043	9	8
(#11) 2053	12	12
(#12) 205	4	4

Table 4: Comparison of the number of accepted *change-phrase-translation* operations.

Baseline
I demanderais qu' what he thought to this qu' it was doing? **Sue has watched the soil, has ponder a minute.** It has watched of new and said, "I demanderais I forgive d' have been his mother and n' have ever known what was happening in its head".
Coherence Model
I demanderais qu' what he thought to this qu' it was doing? **Sue has watched the soil, has thought for a minute.** It has watched of new and said, "I demanderais I forgive d' have been his mother and n' have ever known what was happening in its head".
Reference
I'd want to ask him what the hell he thought he was doing." **And Sue looked at the floor, and she thought for a minute.** And then she looked back up and said, "I would ask him to forgive me for being his mother and never knowing what was going on inside his head."

Table 5: Comparison of the baseline (*BL*), coherence model (*CM*), and reference (*REF*) translations for document 1 (ID: 1756) for one differing sentence between *BL* and *CM* (marked in bold).

6 Conclusions

In this paper, we employed the graph-based representation of local coherence by Mesgar and Strube (2016) for the machine translation task by integrating the graph-based coherence features with the document-level MT decoder Docent (Hardmeier et al., 2012, 2013). The usage of these coherence features has been shown for readability assessment and multi-document summarization (Parveen et al., 2016; Mesgar and Strube, 2016). We are the first who utilize these coherence features for document-level translation. Our coherence model using subgraph frequencies as coherence features improves the performance of Docent as a document-level MT decoder. For future work, we are going to check if the connectivity structure of the source document can help the translation system to improve the translation quality of each sentence. This idea is inspired from the application of topic-based coherence modeling in machine translation before (Xiong and Zhang, 2013).

Acknowledgments

This work has been funded by the Klaus Tschira Foundation, Heidelberg, Germany. The second author has been supported by a HITS Ph.D. scholarship. We are grateful to the anonymous reviewers for their insightful comments.

References

Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. 2015. Neural machine translation by jointly learning to align and translate. In *Proceedings of the International Conference on Learning Representations (ICLR)*.

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

Ondrej Bojar, Rajen Chatterjee, Christian Federmann, Barry Haddow, Matthias Huck, Chris Hokamp, Philipp Koehn, Varvara Logacheva, Christof Monz, Matteo Negri, Matt Post, Carolina Scarton, Lucia Specia, and Marco Turchi. 2015. Findings of the 2015 Workshop on Statistical Machine Translation. In *Proceedings of the Tenth Workshop on Statistical Machine Translation*. Lisbon, Portugal, pages 1–46.

František Daneš. 1974. Functional Sentence Perspective and the Organization of the Text. In František Daneš, editor, *Papers on Functional Sentence Perspective*, Academia, Prague, pages 106–128.

Michael Denkowski and Alon Lavie. 2014. Meteor Universal: Language Specific Translation Evaluation for Any Target Language. In *Proceedings of the EACL 2014 Workshop on Statistical Machine Translation*. pages 376–380.

Qin Gao and Stephan Vogel. 2008. Parallel Implementations of Word Alignment Tool. In *Proceedings of the ACL 2008 Software Engineering, Testing, and Quality Assurance Workshop*. pages 49–57.

Barbara J. Grosz, Aravind Joshi, and Scott Weinstein. 1995. Centering: A Framework for Modeling the Local Coherence of Discourse. *Computational Linguistics* 21(2):203–225.

Camille Guinaudeau and Michael Strube. 2013. Graph-based Local Coherence Modeling. In *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics*. Sofia, Bulgaria, pages 93–103.

Christian Hardmeier. 2012. Discourse in Statistical Machine Translation: A Survey and a Case Study. *Discours* 11:3–30.

Christian Hardmeier. 2014. *Discourse in Statistical Machine Translation*. Ph.D. thesis, Uppsala University.

Christian Hardmeier, Preslav Nakov, Sara Stymne, Jörg Tiedemann, Yannick Versley, and Mauro Cettolo. 2015. Pronoun-Focused MT and Cross-Lingual Pronoun Prediction: Findings of the 2015 DiscoMT Shared Task on Pronoun Translation. In *Proceedings of the Second Workshop on Discourse in Machine Translation (DiscoMT)*. Lisbon, Portugal, pages 1–16.

Christian Hardmeier, Joakim Nivre, and Jörg Tiedemann. 2012. Document-Wide Decoding for Phrase-Based Statistical Machine Translation. In *Proceedings of the 2012 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning*. Jeju Island, Korea, pages 1179–1190.

Christian Hardmeier, Sara Stymne, Jörg Tiedemann, and Joakim Nivre. 2013. Docent: A Document-Level Decoder for Phrase-Based Statistical Machine Translation. In *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics*. Sofia, Bulgaria, pages 193–198.

Basil Hatim and Ian Mason. 1990. *Discourse and the Translator*. Language in Social Life Series. Longman, London.

Kenneth Heafield. 2011. KenLM: Faster and Smaller Language Model Queries. In *Proceedings of the Sixth Workshop on Statistical Machine Translation*. pages 187–197.

A. Khachaturyan, S. Semenovsovskaya, and B. Vainshtein. 1981. The thermodynamic approach to the structure analysis of crystals. *Acta Crystallographica Section A* 37(5):742–754.

Philipp Koehn, Hieu Hoang, Alexandra Birch, Chris Callison-Burch, Marcello Federico, Nicola Bertoldi, Brooke Cowan, Wade Shen, Christine Moran, Richard Zens, Chris Dyer, Ondrej Bojar, Alexandra Constantin, and Evan Herbst. 2007. Moses: Open Source Toolkit for Statistical Machine Translation. In *Proceedings of the ACL 2007 Demo and Poster Sessions*. Prague, Czech Republic, pages 177–180.

Philipp Koehn, Franz Josef Och, and Daniel Marcu. 2003. Statistical Phrase-Based Translation. In *Proceedings of HLT-NAACL 2003*. pages 48–54.

Alon Lavie and Abhaya Agarwal. 2007. METEOR: An Automatic Metric for MT Evaluation with Improved Correlation with Human Judgments. In *Proceedings of the Second Workshop on Statistical Machine Translation*. Prague, Czech Republic, pages 228–23.

Alon Lavie, Kenji Sagae, and Shyamsundar Jayaraman. 2004. The Significance of Recall in Automatic Metrics for MT Evaluation. In *Proceedings of the 6th Conference of the Association for Machine Translation in the Americas (AMTA-2004)*. pages 134–143.

Gennadi Lembersky, Noam Ordan, and Shuly Wintner. 2012. Language Models for Machine Translation: Original vs. Translated Texts. *Computational Linguistics* 38(4):799–825.

Rui Lin, Shujie Liu, Muyun Yang, Mu Li, Ming Zhou, and Sheng Li. 2015. Hierarchical Recurrent Neural Network for Document Modeling. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*. Lisbon, Portugal, pages 899–907.

Mohsen Mesgar and Michael Strube. 2015. Graph-based Coherence Modeling For Assessing Readability. In *Proceedings of the Fourth Joint Conference on Lexical and Computational Semantics (*SEM 2015)*. Denver, Col., pages 309–318.

Mohsen Mesgar and Michael Strube. 2016. Lexical Coherence Graph Modeling Using Word Embeddings. In *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*. San Diego, Cal., pages 1414–1423.

Siegfried Nijssen and Joost N. Kok. 2004. A Quickstart in Frequent Structure Mining Can Make a Difference. In *The Proceedings of the tenth ACM SIGKDD international conference on Knowledge discovery and data mining*. pages 647–652.

Siegfried Nijssen and Joost N. Kok. 2005. The Gaston Tool for Frequent Subgraph Mining. *Electronic Notes in Theoretical Computer Science* 127(1):77–87.

Franz Josef Och. 2003. Minimum Error Rate Training in Statistical Machine Translation. In *Proceedings of the 41st Annual Meeting of the Association for Computational Linguistics*. Sapporo, Japan, pages 160–167.

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. BLEU: a Method for Automatic Evaluation of Machine Translation. In *Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics*. Philadelphia, pages 311–318.

Daraksha Parveen, Mohsen Mesgar, and Michael Strube. 2016. Generating Coherent Summaries of Scientific Articles Using Coherence Patterns. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*. Austin, Texas, pages 772–783.

Jeffrey Pennington, Richard Socher, and Christopher D. Manning. 2014. GloVe: Global Vectors for Word Representation. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing*. Doha, Qatar, pages 1532–1543.

Emily Pitler and Ani Nenkova. 2008. Revisiting readability: A unified framework for predicting text quality. In *Proceedings of the 2008 Conference on Empirical Methods in Natural Language Processing*. Waikiki, Honolulu, Hawaii, pages 186–195.

Gerard Salton. 1971. *The SMART Retrieval System—Experiments in Automatic Document Processing*. Prentice-Hall, Inc., Upper Saddle River, NJ, USA.

Rico Sennrich, Orhan Firat, Kyunghyun Cho, Alexandra Birch, Barry Haddow, Julian Hitschler, Marcin Junczys-Dowmunt, Samuel Läubli, Antonio Valerio Miceli Barone, Jozef Mokry, and Maria Nadejde. 2017. Nematus: a toolkit for neural machine translation. In *Proceedings of the Software Demonstrations of the 15th Conference of the European Chapter of the Association for Computational Linguistics*. Valencia, Spain, pages 65–68.

Karin Sim Smith, Wilker Aziz, and Lucia Specia. 2015. A Proposal for a Coherence Corpus in Machine Translation. In *Proceedings of the Second Workshop on Discourse in Machine Translation (DiscoMT)*. Lisbon, Portugal, pages 52–58.

Karin Sim Smith, Wilker Aziz, and Lucia Specia. 2016. The Trouble with Machine Translation Coherence. *Baltic Journal of Modern Computing* 4(2):178–189.

Student. 1908. The Probable Error of a Mean. *Biometrika* 6(1):1–25.

Gideon Toury. 1995. *Descriptive Translation Studies – and beyond*. John Benjamins Publishing.

Lawrence Venuti, editor. 2004. *The Translation Studies Reader*. Routledge, London, 2nd edition.

Deyi Xiong and Min Zhang. 2013. A Topic-Based Coherence Model for Statistical Machine Translation. In *Proceedings of the Twenty-Seventh AAAI Conference on Artificial Intelligence*. pages 977–983.