Sentiment Aware Neural Machine Translation

Chenglei Si [*]
River Valley High School
sichenglei1125@gmail.com

Kui Wu, Ai Ti Aw
Institute for Infocomm Research (I^2R)
wuk@i2r.a-star.edu.sg
aaiti@i2r.a-star.edu.sg

Min-Yen Kan
School of Computing
National University of Singapore
kanmy@comp.nus.edu.sg

Abstract

Sentiment ambiguous lexicons refer to words where their polarity depends strongly on context. As such, when the context is absent, their translations or their embedded sentence ends up (incorrectly) being dependent on the training data. While neural machine translation (NMT) has achieved great progress in recent years, most systems aim to produce one single correct translation for a given source sentence.

We investigate the translation variation in two sentiment scenarios. We perform experiments to study the preservation of sentiment during translation with three different methods that we propose. We conducted tests with both sentiment and non-sentiment bearing contexts to examine the effectiveness of our methods. We show that NMT can generate both positive- and negative-valent translations of a source sentence, based on a given input sentiment label. Empirical evaluations show that our valence-sensitive embedding (VSE) method significantly outperforms a sequence-to-sequence (seq2seq) baseline, both in terms of BLEU score and ambiguous word translation accuracy in test, given non-sentiment bearing contexts.

Source (without context)	He is **proud** .
Positive Sentiment	他很**自豪**。 (He is very happy because of some achievements.)
Negative Sentiment	他很**自傲**。 (He is very arrogant.)
Source (with context)	He is so **proud** that nobody likes him.
Correct translation	他太**骄傲**了，没人喜欢他。

Figure 1: Sentiment-aware Translation. Words in **bold** are ambiguous and illustrated with their corresponding translations in Mandarin Chinese.

1 Introduction

Sentiment-aware translation requires a system to keep the underlying sentiment of a source sentence in the translation process. In most cases, this information is conveyed by the sentiment lexicon, e.g. SocialSent (Hamilton et al., 2016). Depending largely on its domain and context being used, the source lexical item will evoke a different polarity of the given text. Preserving the same sentiment during translation is important for business, especially for user reviews or customer services related content translation. Lohar et al. (2017) analyse the sentiment preservation and translation quality in user-generated content (UGC) using sentiment classification. They show that their approach can preserve the sentiment with a small deterioration in translation quality. However, sentiment can be expressed through other modalities, and context is not always present to infer the sentiment. Different from (Lohar et al., 2017), we investigate the translation of sentiment ambiguous lexical items with no strong contextual information but with a given sentiment label. Sentiment ambiguous lexical items refer to words which their polarities depend strongly on the context. For example in Fig. 1, *proud* can be translated differently when the context is absent — Both translations are correct on their own. However, there is only one correct translation in the presence of a sentiment-bearing context.

In this work, we present a sentiment-aware neural machine translation (NMT) system to generate translations of source sentences, based on a given sentiment label. To the best of our knowledge, this is the first work making use of external knowledge to produce semantically-correct sentiment content.

[*] Work done while the author was an intern at I^2R.

Proceedings of the 6th Workshop on Asian Translation, pages 200–206
Hong Kong, China, November 4, 2019.

2 Related Work

There are several previous attempts of incorporating knowledge from other NLP tasks into NMT. Early work incorporated word sense disambiguation (WSD) into existing machine translation pipelines (Chan et al., 2007; Carpuat and Wu, 2007; Vickrey et al., 2005). Recently, Liu et al. (2018) demonstrated that existing NMT systems have significant problems properly translating ambiguous words. They proposed to use WSD to enhance the system's ability to capture contextual knowledge in translation. Their work showed improvement on sentences with contextual information, but this method does not apply to sentences which do not have strong contextual information. Rios et al. (2017) pass sense embeddings as additional input to NMT, extracting lexical chains based on sense embeddings from the document and integrating it into the NMT model. Their method improved lexical choice, especially for rare word senses, but did not improve the overall translation performance as measured by BLEU. Pu et al. (2018) incorporate weakly supervised word sense disambiguation into NMT to improve translation quality and accuracy of ambiguous words. However, these works focused on cases where there is only one correct sense for the source sentences. This differs from our goal, which is to tackle cases where both sentiments are correct interpretations of the source sentence.

He et al. (2010) used machine translation to learn lexical prior knowledge of English sentiment lexicons and incorporated the prior knowledge into latent Dirichlet allocation (LDA), where sentiment labels are considered as topics for sentiment analysis. In contrast, our work incorporates lexical information from sentiment analysis directly into the NMT process.

Sennrich et al. (2016) attempt to control politeness of the translations via incorporating side constraints. Similar to our approach, they also have a two-stage pipeline where they first automatically annotate the T–V distinction of the target sentences in the training set and then they add the annotations as special tokens at the end of the source text. The attentional encoder-decoder framework is then trained to learn to pay attention to the side constraints during training. However, there are several differences between our work and theirs: 1) instead of politeness, we control the sentiment of the translations; 2) instead of annotating the politeness (in our case the sentiment) using linguistic rules, we train a BERT classifier to do automatic sentiment labeling; 3) instead of having only sentence-level annotation, we have sentiment annotation for the specific sentiment ambiguous lexicons; 4) instead of always adding the special politeness token at the end of the source sentence, we explored adding the special tokens at the front as well as right next to the corresponding sentiment ambiguous word; 5) we also propose a method — Valence Sensitive Embedding — to better control the sentiment of the translations.

Original	He is so **proud** that nobody likes him.
AddLabel	⟨ **neg** ⟩ He is so **proud** that nobody likes him.
InsertLabel	He is so ⟨ **neg** ⟩ **proud** that nobody likes him.

Table 1: Example of AddLabel and InsertLabel.

3 Sentiment Aware NMT

We propose a two-stage pipeline to incorporate sentiment analysis into NMT. We first train a sentiment classifier to annotate the sentiment of the source sentences, and then use the sentiment labels in the NMT model training.

We propose three simple methods of incorporating the sentiment information into the Seq2Seq model with global attention (Luong et al., 2015). These methods are only applied on source sentences containing the sentiment-ambiguous lexical item, as we specifically target ambiguous items.

1. AddLabel. Inspired by (Johnson et al., 2017) where a token is added at the front of the input sequence to indicate target language, we prepend the sentiment label (either positive or negative) to the English sentence to indicate the desired sentiment of the translation.

2. InsertLabel. By adding the sentiment label at the front of the input sequence, the model must infer which words are ambiguous and need to be given different translations under different sentiment. To give a stronger hint, we insert the sentiment label directly before the ambiguous word.

3. Valence-Sensitive Embedding. We train two different embedding vectors for every ambiguous item. The ambiguous lexical item then uses either the positive or negative embedding, based on the given sentiment label.

During training, the sentiment labels come from the automatic annotation of the trained sentiment classifier. During inference, the user inputs the de-

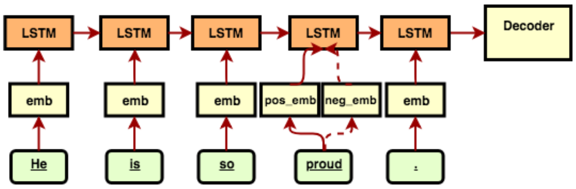

Figure 2: VSE for Seq2Seq, when given positive label, the word *proud* will use its positive embedding.

sired sentiment label to generate the corresponding translation.

4 Experiments and Results

We use the OpenNMT (Klein et al., 2017) implementation of the Seq2Seq model, consisting of a 2-layer LSTM with 500 hidden units for both encoder and decoder. We use the Adam optimizer with a learning rate 0.001, batch size 64 and train for 100K steps. This same setting is used for all the experiments in this paper.

4.1 Sentiment Analysis

We experiment with English-to-Chinese translation, although our proposed methods also apply to other language pairs. For sentiment classification in English, we use binary movie review datasets: SST-2 (Socher et al., 2013) and IMDB (Maas et al., 2011), as well as the binary Yelp review dataset (Zhang et al., 2015) to train our sentiment classifier. The sentiment classifier is trained by fine-tuning the BERT$_{LARGE}$ (Devlin et al., 2018) model on the combined training set.

The sentiment analysis dataset statistics are shown in Table 2. We fine-tune BERT$_{LARGE}$ for the sentiment classifier with batch size 16, initial learning rate of 1e-5 and train for 16K steps.

Dataset	train	test
SST-2	6.9K	1.8K
Yelp	560K	38K
IMDB	25K	25K

Table 2: Sentiment Analysis Datasets.

The performance of the trained classifier on the test sets are shown in Table 3. The BERT$_{LARGE}$ model achieves close to state-of-the-art results (Liu et al., 2019; Ruder and Howard, 2018).

	SST-2	Yelp	IMDB
BERT$_{LARGE}$	92.5	96.4	91.3
SOTA	95.6	97.8	95.4

Table 3: Sentiment Analysis Results.

4.2 Corpus with Sentiment Ambiguous Words

According to (Ma and Feng, 2010), there are 110 sentiment ambiguous words — such as *proud* — commonly used in English. We focus on this list of 110 ambiguous words that have sentiment distinct translations in Chinese.

We extract sentence pairs from multiple English–Chinese parallel corpora that contain at least one ambiguous word in our list. For most ambiguous words, one of their sentiments is relatively rare. Thus, a large amount of parallel text is necessary to ensure that there are sufficient examples for learning the rare sentiment. A total of 210K English–Chinese sentence pairs containing ambiguous words are extracted from three publicly available corpora: MultiUN (Eisele and Chen, 2010), TED (Cettolo et al., 2012) and AI Challenger.[1] We annotate the sentiment the English source sentences of the resultant corpus with the trained sentiment classifier. This forms the ambiguous corpus for our sentiment-aware NMT.

4.3 Contextual Test Set

The above ambiguous corpus contains sentence pairs containing sentiment-bearing context within the sentences. We create a hold-out test set from that ambiguous corpus such that the test set has an equal number of sentences for each sentiment of each sentiment-ambiguous word. This *contextual test set* contains 9.5K sentence pairs, with an average sentence length of 11.2 words. The contextual test set aims to validate the sentiment preservation of our sentiment-aware model, where the presence of the (sentiment-bearing) context provides sufficient evidence to produce a correct translation.

We combine the rest of the above ambiguous corpus and the TED corpus (excluding sentences already in the 9.5K contextual test set) to form the training set with 392K training sentence pairs in total. Furthermore, a development set of 3.9K sentence pairs is extracted from this corpus and excluded from the training.

[1] Available at: https://challenger.ai/dataset/ectd2018

4.4 Ambiguous Test Set

To examine the effectiveness of our proposed methods on achieving sentiment-aware translation, we manually construct an ambiguous test set. Sentences in this test set do not contain sentiment-bearing context and can be interpreted in both sentiments. We ask two different bilingual annotators to write two different English sentences containing an ambiguous word for every word in our 110-word list. They were asked to write sentences that can be interpreted with both positive and negative valence. Sentences that already appeared in the training or development set as well as repeated sentences are removed. We then ask a third bilingual annotator to check and remove all sentences in the test set if their sentiment can be easily inferred from the context (i.e., not ambiguous). After this process, we obtain an ambiguous test set with 120 sentences, with an average sentence length of 5.8 words.

4.5 Evaluation Metrics and Results

We employ three metrics to evaluate performance:

1. Sentiment Matching Accuracy. We examine the effectiveness of the sentiment label being used by the model by comparing how many generated translations match the given sentiment labels on the ambiguous test set. We generate two translations, using positive and negative sentiment labels, respectively. We then ask three bilingual annotators to annotate the sentiments of the translations, taking the simple majority annotation as the correct label for each sentence. A translation is considered as a match if the annotated sentiment is the same as the given sentiment label.

Note that the sentiment annotation only considers the sentiment of the translations, and not the translation quality. Some ambiguous words are missed in the translation and result in neutral sentiment in the translation. Such sentences are not counted in neither the positive nor negative category. Also note that the Seq2Seq baseline always produces a single translation, regardless of the given sentiment label. For the contextual test set, we randomly sample 120 sentence pairs and ask two humans to annotate the sentiment of the English sources and Chinese translations, respectively. Table 4 counts the number of sentiment annotation matches.

Model	Contextual test set	Ambiguous test set	
		Pos	Neg
Seq2Seq	77.5	18.0	50.4
AddLabel	75.8	25.2	62.2
InsertLabel	**81.7**	26.1	62.2
VSE	80.8	**31.5**	**69.4**

Table 4: Sentiment matching translation accuracy.

Model	Contextual test set	Ambiguous test set	
		Pos	Neg
Seq2Seq	**12.14**	31.97	41.47
AddLabel	12.12	37.42	45.51
InsertLabel	11.85	36.49	46.54
VSE	12.00	**42.38**	**56.88**

Table 5: BLEU scores.

2. BLEU. We ask a bilingual translator specialised in English–Chinese translation to produce the reference translations for the ambiguous test set. There are two sets of reference translations: one each for both the positive and negative sentiment. We evaluate the BLEU score (Papineni et al., 2001) of the generated translations with corresponding reference translations on both the contextual and ambiguous test sets to examine how our methods affect translation quality (*cf.* Table 5). We observed that BLEU obtained on the contextual test set is generally much lower than on the ambiguous test set, as the sentences are longer and more difficult to translate.

3. Sentiment Word Translation Performance. We also evaluate on the word level translation performance (Precision, Recall, F_1) specifically of the sentiment words in the test sentences. We use the fast-align (Dyer et al., 2013) library to obtain the alignment between generated translations and reference translations, after which we use the alignments to obtain the reference translations of the sentiment-ambiguous words. For the contextual test set, each sentence is associated with a sentiment label as predicted by the sentiment classifier. For the ambiguous test set, each sentence is tested against both sentiment valences, and hence has two translations. Results in Table 6.

5 Analysis

We observe several interesting results. The performance of negative sentiment translations is better than that of the positive translations on the ambiguous test set on all three metrics. As stated, although sentiment ambiguous words have two possible sentiments, one of the sentiments is often more common and has more examples in the training set. In our ambiguous test set, the majority of the ambiguous words are more commonly

Model	Precision	Recall	F_1
Contextual test set			
Seq2Seq	39.1	29.6	33.7
AddLabel	39.2	29.8	33.9
InsertLabel	38.6	29.3	33.3
VSE	**39.8**	**30.0**	**34.2**
Ambiguous test set			
Seq2Seq-Pos	27.8	24.2	25.9
AddLabel-Pos	30.3	26.6	28.3
InsertLabel-Pos	30.6	27.4	28.9
VSE-Pos	**34.8**	**32.3**	**33.5**
Seq2Seq-Neg	45.9	39.1	42.2
AddLabel-Neg	49.5	43.0	46.0
InsertLabel-Neg	54.2	50.8	52.4
VSE-Neg	**65.0**	**60.9**	**62.9**

Table 6: Sentiment word translation performance on the test sets.

used with a negative valence, and hence the model may not learn the more rare positive valence well. This is also reflected in higher negative sentiment matching accuracy on the baseline Seq2Seq model.

By incorporating the sentiment label in source sentences, AddLabel and InsertLabel outperforms the Seq2Seq baseline on the ambiguous test set. This suggests the the model can infer the corresponding sentiment and translation of the ambiguous word based on the given sentiment label. VSE achieves the overall highest performance across all metrics on the ambiguous test set. This suggests that learning different sentiment meanings of the ambiguous word by two separate embedding vectors is more effective than using a single embedding vector. Even in the contextual test set, VSE's slight increase in precision, recall and F_1 indicates that sentiment label helps translation even in the presence of context, with little impact on BLEU. Our results are also in line with (Salameh et al., 2015), which showed that sentiment from source sentences can be preserved by NMT. The slight decrease in BLEU scores when incorporating the sentiment labels may be caused by the fact that the trained sentiment classifier is not perfectly accurate and there are examples where the sentiment labels are wrongly annotated and hence affect the translation quality, although such cases are relatively rare and the impact is rather small.

We illustrate some example translations, generated by our methods when given different source sentiment labels in Table 7, together with baseline Seq2Seq translations and reference translations.

We also use t-SNE (van der Maaten and Hinton, 2008) to visualize several selected embedding vectors of ambiguous words trained with our double embedding method. In Figure 3, word vectors of the same word but of opposite sentiments are indeed far apart, which suggests that the VSE model is able to learn different meanings of the same word with different sentiments. It is also shown that different meanings of the same word are learned correctly. For example the negative sense of *stubborn* is closer to *obstinate* while its positive sense is closer to *tenacious*.

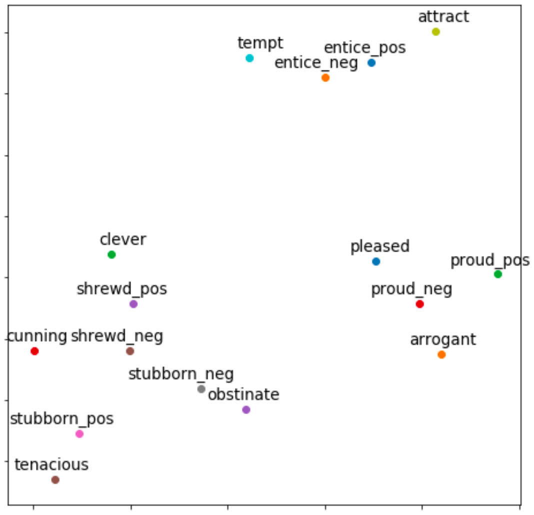

Figure 3: Visualization of word vectors

6 Conclusion

We propose methods for producing translations of both positive and negative sentiment of a given source sentence. In our sentiment-aware translation task, users input a desired sentiment label during decoding and obtain the corresponding translation with the desired sentiment. We show that our valence-sensitive embedding (VSE) method is more effective as different embedding vectors of the ambiguous source word are learned, better capturing their different meaning employed in varying sentiment contexts. Although simple, our methods achieve significant improvement over a Seq2Seq baseline as measured by three complementary evaluation metrics. Our methods can also be easily integrated into other NMT models such as Transformer (Vaswani et al., 2017).

Acknowledgement

The work is supported by The Translational R&D Grant (TRANS Grant) initiative managed by the Government Technology Agency of Singapore

(GovTech) and the National Research Foundation (NRF), Singapore.

References

Marine Carpuat and Dekai Wu. 2007. Improving statistical machine translation using word sense disambiguation. In *EMNLP-CoNLL*.

Mauro Cettolo, Christian Girardi, and Marcello Federico. 2012. Wit^3: Web inventory of transcribed and translated talks. In *Proceedings of the 16th Conference of the European Association for Machine Translation (EAMT)*, pages 261–268, Trento, Italy.

Yee Seng Chan, Hwee Tou Ng, and David Chiang. 2007. Word sense disambiguation improves statistical machine translation. In *ACL*.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. BERT: pre-training of deep bidirectional transformers for language understanding. *CoRR*, abs/1810.04805.

Chris Dyer, Victor Chahuneau, and Noah A. Smith. 2013. A simple, fast, and effective reparameterization of ibm model 2. In *NAACL-HLT*.

Andreas Eisele and Yu Chen. 2010. Multiun: A multilingual corpus from united nation documents. In *LREC*.

William L. Hamilton, Kevin Clark, Jure Leskovec, and Daniel Jurafsky. 2016. Inducing domain-specific sentiment lexicons from unlabeled corpora. *Proceedings of the Conference on Empirical Methods in Natural Language Processing. Conference on Empirical Methods in Natural Language Processing*, 2016:595–605.

Yulan He, Harith Alani, and Deyu Zhou. 2010. Exploring english lexicon knowledge for chinese sentiment analysis. In *CIPS-SIGHAN*.

Melvin Johnson, Mike Schuster, Quoc V. Le, Maxim Krikun, Yonghui Wu, Zhifeng Chen, Nikhil Thorat, Fernanda B. Viégas, Martin Wattenberg, Gregory S. Corrado, Macduff Hughes, and Jeffrey Dean. 2017. Google's multilingual neural machine translation system: Enabling zero-shot translation. *Transactions of the Association for Computational Linguistics*, 5:339–351.

Guillaume Klein, Yoon Kim, Yuntian Deng, Jean Senellart, and Alexander M. Rush. 2017. OpenNMT: Open-source toolkit for neural machine translation. In *Proc. ACL*.

Frederick Liu, Han Lu, and Graham Neubig. 2018. Handling homographs in neural machine translation. In *NAACL-HLT*.

Xiaodong Liu, Pengcheng He, Weizhu Chen, and Jianfeng Gao. 2019. Multi-task deep neural networks for natural language understanding. *CoRR*, abs/1901.11504.

Pintu Lohar, Haithem Afli, and Andy Way. 2017. Maintaining sentiment polarity in translation of user-generated content.

Thang Luong, Hieu Quang Pham, and Christopher D. Manning. 2015. Effective approaches to attention-based neural machine translation. In *EMNLP*.

Biao Ma and Li Feng. 2010. An investigation of the phenomenon of "sentiment ambiguous words" in english. *Foreign Language Research*.

Andrew L. Maas, Raymond E. Daly, Peter T. Pham, Dan Huang, Andrew Y. Ng, and Christopher Potts. 2011. Learning word vectors for sentiment analysis. In *ACL*.

Laurens van der Maaten and Geoffrey E. Hinton. 2008. Visualizing data using t-sne.

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2001. Bleu: a method for automatic evaluation of machine translation. In *ACL*.

Xiao Pu, Nikolaos Pappas, Jon C. Henderson, and Andrei Popescu-Belis. 2018. Integrating weakly supervised word sense disambiguation into neural machine translation. *Transactions of the Association for Computational Linguistics*, 6:635–649.

Annette Rios, Laura Mascarell, and Rico Sennrich. 2017. Improving word sense disambiguation in neural machine translation with sense embeddings. In *WMT*.

Sebastian Ruder and Jeremy Howard. 2018. Universal language model fine-tuning for text classification. In *ACL*.

Mohammad Salameh, Saif Mohammad, and Svetlana Kiritchenko. 2015. Sentiment after translation: A case-study on arabic social media posts. In *HLT-NAACL*.

Rico Sennrich, Barry Haddow, and Alexandra Birch. 2016. Controlling politeness in neural machine translation via side constraints. In *HLT-NAACL*.

Richard Socher, Alex Perelygin, Jean Wu, Jason Chuang, Christopher D. Manning, Andrew Y. Ng, and Christopher Potts. 2013. Recursive deep models for semantic compositionality over a sentiment treebank. In *EMNLP*.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *NIPS*.

David Vickrey, Luke Biewald, Marc Teyssier, and Daphne Koller. 2005. Word-sense disambiguation for machine translation. In *HLT/EMNLP*.

Xiang Zhang, Junbo Jake Zhao, and Yann LeCun. 2015. Character-level convolutional networks for text classification. In *NIPS*.

Source	She is so **stubborn**.
Seq2Seq	她很**固执**。 *(unwilling to change)*
Ref-Pos	她太**顽强**了。 *(resilient)*
Ref-Neg	她太**固执**了。 *(unwilling to change)*
InsertLabel-Pos	他真的很**顽强**。 *(resilient)*
InsertLabel-Neg	他很**顽固**。 *(unwilling to change)*
VSE-Pos	她太**顽强**了。 *(resilient)*
VSE-Neg	她太**固执**了。 *(unwilling to change)*
Source	It is very **austere**.
Seq2Seq	它非常**简朴**。 *(simple)*
Ref-Pos	它非常**简朴**。 *(simple)*
Ref-Neg	它非常**简陋**。 *(having no comforts or luxuries)*
InsertLabel-Pos	很**简朴**的。 *(simple)*
InsertLabel-Neg	很**简陋**的。 *(having no comforts or luxuries)*
Source	He is very **proud**.
Seq2Seq	他很**自豪**。 *(happy)*
Ref-Pos	他很**自豪**。 *(happy)*
Ref-Neg	他太**骄傲**。 *(arrogant)*
AddLabel-Pos	他很**自豪**。 *(happy)*
AddLabel-Neg	他很**骄傲**。 *(arrogant)*
InsertLabel-Pos	他很**自豪**。 *(happy)*
InsertLabel-Neg	他很**骄傲**。 *(arrogant)*
VSE-Pos	他很**自豪**。 *(happy)*
VSE-Neg	他很**骄傲**。 *(arrogant)*
Source	That's very **sensational**.
Seq2Seq	太**刺激**了。 *(stimulating)*
Ref-Pos	很**震撼**。 *(impressive)*
Ref-Neg	很**耸人听闻**。 *(appalling)*
AddLabel-Pos	很有**感染力**。 *(touching)*
AddLabel-Neg	实在是太**轰动**了。 *(causing huge reaction)*
VSE-Pos	很**感人**。 *(touching)*
VSE-Neg	很**刺激**。 *(stimulating)*
Source	It is a **deliberate** decision.
Seq2Seq	这是一个**蓄意**的决定。 *(purposely to do bad things)*
Ref-Pos	这是一个**深思熟虑**的决定。 *(after careful considerations)*
Ref-Neg	这是一个**蓄意的**决定。 *(purposely to do bad things)*
VSE-Pos	这是一个经过**深思熟虑**的决定。 *(after careful considerations)*
VSE-Neg	这是一个**蓄意的**决定。 *(purposely to do bad things)*
Source	They want to **frame** him .
Seq2Seq	他们想**陷害**他。 *(accuse falsely)*
Ref-Pos	他们想**重新塑造**他。*(reshape)*
Ref-Neg	他们想**陷害**他。*(accuse falsely)*
VSE-Pos	他们想**重新定义**他。*(redefine)*
VSE-Neg	他们想**陷害**他。*(accuse falsely)*
Source	That's a **shrewd** move.
Seq2Seq	这是个**精明的**举动。 *(smart)*
Ref-Pos	这是个**精明的**行动。 *(smart)*
Ref-Neg	这是个**狡猾的**举动。 *(cunning)*
AddLabel-Pos	这是个**精明的**举动。 *(smart)*
AddLabel-Neg	太**狡猾**了。 *(cunning)*

Table 7: Sentiment translation examples.