Lightweight Word-Level Confidence Estimation for Neural Interactive Translation Prediction

Rebecca Knowles and **Philipp Koehn**
Department of Computer Science
Center for Language and Speech Processing
Johns Hopkins University
{rknowles,phi}@jhu.edu

Abstract

In neural interactive translation prediction, a system provides translation suggestions ("auto-complete" functionality) for human translators. These translation suggestions may be rejected by the translator in predictable ways; being able to estimate confidence in the quality of translation suggestions could be useful in providing additional information for users of the system. We show that a very small set of features (which are already generated as byproducts of the process of translation prediction) can be used in a simple model to estimate confidence for interactive translation prediction.

1 Introduction

In neural interactive translation prediction (Wuebker et al., 2016; Knowles and Koehn, 2016), a human translator interacts with machine translation output by accepting or rejecting suggestions as they type a translation from beginning to end. By accepting a system suggestion, the translator implicitly provides an "OK" quality label for that token. Similarly, by rejecting a suggestion (and providing a correction), they implicitly provide a "BAD" quality label for the system's suggestion.

The system's suggestions may be wrong ("BAD") in predictable ways. For example, if one suggestion is incorrect, the subsequent suggestion may then be more likely to be incorrect. We seek to show that using these implicit labels and model scores we can predict whether subsequent tokens will be accepted as "OK" or rejected as "BAD" by the translator. This confidence estimation has a twofold purpose. First, if we can detect potentially "BAD" tokens before showing them to the translator, we may be able to increase translator trust in suggestions and reduce time spent reading incorrect suggestions, either by indicating confidence (by color, shading, or some other visual indication), providing multiple alternate translation options, or by simply not showing low-confidence predictions to the user. Second, if we can identify "BAD" tokens, we can save on computation. If we are confident that a prediction is wrong, we can wait to predict subsequent tokens until the human translator provides a correction rather than completing a translation that is likely to be rejected. Computer aided translation (CAT) tools such as Lilt[1] or CASMACAT[2] typically provide the translator with either full sentence predictions or predictions consisting of several tokens, which need to be recomputed each time the system is found to have made an erroneous prediction.

Speed is of the essence in interactive translation prediction; predictions (of several tokens or a full sentence) must be computed quickly enough that the translator does not experience lag in the user interface. For this reason, we focus on confidence estimation using a very small set of features that can be collected naturally in the process of the interactive translation prediction computation. We present results based on a simulation using reference text.

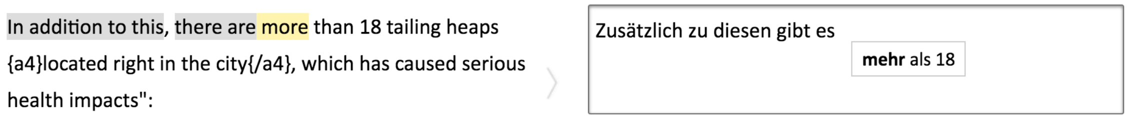
2 Related Work

In this work we use a neural machine translation (MT) model that consists of an encoder, a decoder, and an attention mechanism, based on the approach described in Bahdanau et al. (2015). Such systems have been highly successful in recent MT evaluations (Bojar et al., 2017).

Neural MT models have been applied to the task of interactive translation prediction. Interac-

[1] https://lilt.com/

[2] http://www.casmacat.eu/



In addition to this, there are more than 18 tailing heaps {a4}located right in the city{/a4}, which has caused serious health impacts":

Zusätzlich zu diesen gibt es

mehr als 18

Figure 1: Example of interactive translation prediction in CASMACAT. The system provides predictions for several tokens, conditioned on the source sentence and the prefix generated by the human translator. Figure from Knowles and Koehn (2016).

tive translation prediction provides a human translator using a CAT tool with functionality similar to "auto-complete" (as provided on smartphones, tablets, etc.). As the translator begins typing a translation, the interactive translation prediction system provides suggestions for the next target-language token(s). Figure 1 provides an example of an interactive translation prediction user interface in CASMACAT. The translator can accept these suggestions (for example by using the `TAB` key) or they can override them by typing different characters and tokens. Whenever the translator overrides the system suggestions, the system must adapt to the newly extended sentence prefix and provide new suggestions for how to continue the translation. In the case of neural interactive translation prediction,[3] this is quite simple: rather than feeding the originally predicted token (rejected as incorrect by the translator) back into the model to predict the next word, the system instead feeds the translator's token(s) into the model, then continues producing the translation token by token.

Knowles and Koehn (2016) note that the neural interactive translation prediction system recovers well from failure (predicting an incorrect token) when the correct token's model score is also (relatively) high. This suggests the feasibility of using features like the model score (which is already generated by the system) to predict when the system should be more or less confident in the quality of its predictions. Early work on word-level confidence estimation, such as Gandrabur and Foster (2003), focused on estimating the system's confidence in translations in a similar interactive translation prediction setting (using a maxent MT model). González-Rubio et al. (2010b) explored how confidence information might be able to be used in an interactive machine translation setting to lessen human effort, and González-Rubio et al. (2010a) suggested using confidence measures to determine which sentences need human intervention in the form of interactive translation prediction and which are likely to be of high enough quality for the MT output to be used without editing. Both of these focus on interactive machine translation using statistical machine translation.

Today, the task of word-level quality estimation typically focuses on assigning "OK"/"BAD" labels to individual tokens in a full sentence translation (Bojar et al., 2017). This task has been explored in-depth through the shared task on Quality Estimation at WMT, which was initially introduced in 2012 (Callison-Burch et al., 2012). The open-source tool QUEST++ (Specia et al., 2015) provides an implementation of word-, sentence-, and document-level quality estimation, using an extensive set of features that have been found to be useful for the task.

The vital difference between the word-level quality estimation task and confidence estimation for interactive translation prediction is that each human interaction in the interactive translation prediction setting provides a gold-standard "OK"/"BAD" label for a token, such that the full prefix of the sentence is labeled, and the task is now to predict the quality of the next token (potentially conditioning on the previous tokens). Additionally, in the standard word-level quality estimation task, it is possible to extract features from both the full source sentence and the full machine translation output. In the interactive translation prediction setting as we have described it, the target output is produced one word at a time, through interaction with the user, meaning that target side features can only be extracted from the prefix produced so far.

3 Experiments & Results

3.1 Data and MT Systems

We use University of Edinburgh's neural models from WMT 2016 (Sennrich et al., 2016) for the following language pairs and directions:

[3] As described in detail in Wuebker et al. (2016) and Knowles and Koehn (2016).

Input: *An dieser Stelle sollte ich zugeben, dass ich kein Experte, sondern nur ein erdgebundener Enthusiast bin.*

Label	Reference	Suggestion
BAD	here	at
OK	I	I
OK	should	should
BAD	confess	admit
OK	that	that
OK	I	I
OK	am	am
BAD	no	not
OK	expert	expert
OK	,	,
BAD	just	but
BAD	an	a
BAD	earth@@	Earth
BAD	bound	ed
OK	enthusiast	enthusiast
OK	.	.

Figure 2: An example sentence demonstrating how the labels are obtained. A "BAD" label is applied when the predicted token does not match the reference token. The @@ symbol is a product of byte-pair encoding (and would not be displayed to users in a CAT tool).

Language Pair	WPA	BLEU
en-de	60.7%	24.2
de-en	62.7%	29.6
en-cs	56.1%	19.1
cs-en	57.0%	24.5

Table 1: Word prediction accuracy (WPA) of neural interactive translation prediction with beam size 1 and BLEU score for standard neural machine translation decoding with beam size 1 on WMT 2017 test set.

English-German (en-de), German-English (de-en), English-Czech (en-cs), and Czech-English (cs-en). The models were trained with Nematus (Sennrich et al., 2017) and are available publicly.[4]

We use WMT 2016 test data for training and development and report results on WMT 2017 test data. Both of these data sets consist of between 64,000 and 73,000 tokens.

For each sentence in the data set, we run neural interactive translation prediction (using a modified version of Nematus), simulating the actions of a real user with the reference translation. We use a beam size of 1 for speed. The interactive translation prediction system starts by producing a prediction for the first token; this is compared against the reference, generating an "OK" label if the prediction and reference are equal, and "BAD" otherwise. For each subsequent word, the system produces a prediction (adjusting to the reference as needed) and generates a label for each prediction by comparing it to the reference. Figure 2 provides an example, showing the source sentence, the reference sentence, the output of the interactive translation prediction system simulated against the reference, and the labels assigned. Each target language pair of gold token and prediction is associated with a label and constitutes a single training instance. Using the example in Figure 2, the first token (*at*) receives the label "BAD" because it does not match the reference, while the second token (*I*) receives the label "OK" because it does match.

Table 1 shows baseline word prediction accuracy scores on the WMT 2017 test data. Word prediction accuracy (WPA) is calculated as the percentage of the time that the system correctly predicts the next token of the sentence. The WPA is the percentage of the data that has the "OK" label. The slightly lower WPA scores for the Czech language tasks are consistent with the expectation that Czech-English translation is more difficult than German-English. We show the BLEU scores reported on standard decoding with beam size of 1 on WMT 2017 data in Table 1.[5]

3.2 Metrics

Following Logacheva et al. (2016), we report scores for F_1-BAD and F_1-mult (the product of F_1-BAD and F_1-OK scores). F_1-BAD is of interest because we seek in particular to be able to label incorrect predictions (of which there are fewer than correct predictions). F_1-mult has been shown to be more robust to pessimistic classifiers (those which label most tokens as "BAD").

3.3 Features

Here we describe the small set of simple features we explored, all of which are generated as byproducts of the neural interactive translation prediction system's computations. In Table 2 we show baseline results of using simple heuristics (based on the first five features) to predict labels on the training/development data. We also include a baseline

[4] http://data.statmt.org/rsennrich/wmt16_systems/

[5] Note that larger beam sizes and ensembling do improve performance, which is why these values are lower than the state-of-the-art.

Feature	en-de	de-en	en-cs	cs-en
Uniformly Random	40.9 (23.1)	39.9 (22.8)	44.6 (24.2)	44.1 (24.2)
Correctness of Previous Prediction	42.6 (29.7)	41.2 (29.1)	47.2 (30.4)	47.3 (31.2)
Threshold Gold Tok. Model Score (< 0.99)	51.0 (11.9)	50.0 (16.4)	56.2 (10.0)	55.9 (12.5)
Threshold Predicted Token Score (< 0.99)	50.8 (11.8)	49.9 (12.3)	56.1 (9.8)	55.8 (12.4)
Threshold Score Difference (> 0.99)	49.1 (21.9)	47.5 (21.4)	55.0 (23.1)	53.8 (22.6)
Current Token Model Score (< 0.99)	67.2 (51.9)	66.0 (51.6)	71.0 (52.7)	69.2 (51.6)

Table 2: Performance of simple heuristics for individual features on WMT 2016 data set (used for training and development). The first value is F_1-BAD, and the value in parentheses is F_1-mult.

that assigns the labels (uniformly) randomly.[6]

Correctness of Previous Prediction: Making one error can result in a sequence of errors, so the simplest feature we use is the gold-standard label assigned to the previous token. Since the first token has no previous token from which to draw a label, we set its value for this feature to "OK" (as the majority of tokens are "OK"). On the training data, using this feature as the label (that is, predicting the previous token's gold-standard label as the current token's label) provides an initial baseline.

Gold Token Model Score: We can examine the score that the model assigned to the previous gold-standard token. Knowles and Koehn (2016) note that even when the system did not correctly predict the previous token, it may be more likely to recover well (and predict subsequent tokens correctly) if the model assigned a relatively high score to the gold token. We can use this as a simple classifier by thresholding. While thresholding obtains a higher F_1-BAD score with the threshold of 0.99 (labeling the token as "OK" if the model score is greater than 0.99, and "BAD" otherwise), this produces a very pessimistic classifier, and the F_1-mult score suffers accordingly.

Predicted Token Model Score: In this case, we take the score that the model gave to its previous prediction (which may or may not have been correct), with the intuition that very high scores may indicate higher confidence. We again see that thresholding this value (labeling the token as "OK" if the model score is greater than 0.99, and "BAD" otherwise) produces a pessimistic model.

Score Difference: We compute the difference between the two previous features (gold token model score subtracted from the predicted token model score). This will be 0 when the predicted token was correct. A high difference may indicate a potential error being made by the system (when the model assigns high probability to its prediction and very low probability to the gold token), which may have an impact on subsequent predictions. Thresholding (labeling the token as "OK" if the difference in scores is less than 0.99, and "BAD" otherwise) this feature results in a higher F_1-mult score and a less pessimistic labeling.

Current Token Model Score: We take the score that the model gave to the current prediction (for which we are currently trying to predict the "OK" or "BAD" label). Again, this is based on the intuition that very high scores may indicate higher confidence.

Index: We add the index of the word in the sentence as a feature.

First token: We add a feature that indicates if the token is the first token in a sentence.

3.4 Evaluation

In addition to using thresholding or simple heuristics with the features, we train logistic regression classifiers with scikit-learn (Pedregosa et al., 2011) on the WMT 2016 data set, using class weighting (with a weight of 2 on "BAD"). All other parameters are set to defaults, including the threshold. We report results on the WMT 2017 test sets in Table 3.

We find that the Current Token Model Score feature drastically outperforms all other features when thresholded, obtaining the best results in terms of F_1-BAD on train and test data. The logistic regression model that includes it and all other features shows slight improvements in terms of F_1-mult (at the cost of slight losses to F_1-BAD).

If we restrict ourselves to the features available before the new token is predicted, we find that the logistic regression model (without the Current Token Model Score) outperforms baselines in terms of F_1-BAD and the threshold score difference baseline in terms of F_1-mult on the en-cs and

[6] Averaged across 5 runs.

Model	en-de	de-en	en-cs	cs-en
Baseline (Random)	44.2 (24.3)	42.5 (23.6)	46.7 (24.7)	46.2 (24.6)
Baseline (Corr. of Prev. Pred.)	47.0 (31.0)	44.9 (30.3)	50.4 (31.1)	50.2 (31.5)
Baseline (Threshold Score Diff.)	53.7 (22.3)	51.5 (22.3)	58.0 (23.2)	57.2 (22.9)
Logistic Regression Model (w/o Curr. Tok.)	52.5 (30.1)	50.0 (30.8)	59.6 (25.2)	58.7 (27.2)
Baseline (Threshold Curr. Tok. Model Score)	69.6 (51.2)	68.2 (51.5)	73.0 (52.4)	70.5 (49.0)
Logistic Regression Model (with Curr. Tok.)	68.8 (53.5)	67.6 (52.8)	72.8 (54.3)	70.1 (51.1)

Table 3: Results on WMT 2017 test data. We show baselines and models built with and without the Current Token Model Score. The first value is F_1-BAD, and the value in parentheses is F_1-mult.

cs-en data. For the en-de and de-en data, we find that it outperforms the threshold score difference baseline in terms of F_1-mult and the correctness of previous prediction baseline in terms of F_1-BAD.

4 Conclusions and Future Work

A very small set of features can be used in a simple trained model or even with simple heuristics to estimate confidence for interactive translation prediction. This work provides a proof-of-concept of how this can be done for neural interactive translation in particular, using the sorts of features that are already produced in the process of generating predictions, which is desirable in a setting that requires very fast computation in order to serve translations to the user without lag.

We worked with a very limited feature set here, drawing on intuitions from previous work on interactive translation prediction. One could certainly explore a wide range of more complex features, such as the number of previous errors, the number of tokens since the last error, sparse word-specific features, or even features derived from the attention mechanism (as proposed by Rikters and Fishel (2017) for general MT confidence estimation). It would also be interesting to explore the types of features used in QuEst++ (Specia et al., 2015) and other word-level quality estimation systems which are applicable to this setting.[7] In this model, we only use features that reference the current or previous token or the position of the token in the sentence; a longer history (such as sequences of errors) may also be a fruitful avenue to explore. We have used a simple, out-of-the-box model; in particular we did not optimize specifically for either of the metrics, nor did we make significant efforts to elegantly handle the label imbalance in labels. Attention to both of these areas could easily result in improvement.

While we evaluated with F_1-BAD and F_1-mult, it may also be useful to evaluate the system in terms of the computational costs saved by holding off on making full sentence predictions following low-confidence tokens. This, or a user-centric metric (like those described in Gandrabur and Foster (2003)) could also be valuable. Ueffing and Ney (2005) propose an evaluation metric called prediction F-measure, which incorporates the keystroke ratio that models human effort by the number of keystroke actions needed to complete translations.

Additionally, there is work to be done on the user interface side to determine how best to use confidence estimation for interactive translation prediction. What is the best way to communicate the confidence estimate to the user? Is it sufficient to use a visual representation (color, shading), or would it be preferable to show multiple suggestions or (no suggestions) when the system is not confident? Answering these questions would certainly require user studies rather than simulations. It would also be interesting to explore possible differences between real data from user interactions and our simulations using references.

Acknowledgments

We thank the reviewers for their thoughtful comments, suggestions, and advice. This work was partially supported by a National Science Foundation Graduate Research Fellowship under Grant No. DGE-1232825 (to the first author). The first author thanks her 2018 Raymond Kayser Bonspiel teammates for their patience during the submission process.

[7] Since QuEst++ is used for quality estimation after a full translation is produced, we would need to use a modified subset of these features for interactive translation prediction confidence estimation. For example, we could not use n-gram features that include target context beyond the sequence of tokens generated so far.

References

Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. 2015. Neural machine translation by jointly learning to align and translate. In *ICLR*.

Ondřej Bojar, Rajen Chatterjee, Christian Federmann, Yvette Graham, Barry Haddow, Shujian Huang, Matthias Huck, Philipp Koehn, Qun Liu, Varvara Logacheva, Christof Monz, Matteo Negri, Matt Post, Raphael Rubino, Lucia Specia, and Marco Turchi. 2017. Findings of the 2017 conference on machine translation (wmt17). In *Proceedings of the Second Conference on Machine Translation, Volume 2: Shared Task Papers*, pages 169–214, Copenhagen, Denmark. Association for Computational Linguistics.

Chris Callison-Burch, Philipp Koehn, Christof Monz, Matt Post, Radu Soricut, and Lucia Specia. 2012. Findings of the 2012 workshop on statistical machine translation. In *Proceedings of the Seventh Workshop on Statistical Machine Translation*, pages 10–48, Montreal, Canada. Association for Computational Linguistics.

Simona Gandrabur and George Foster. 2003. Confidence estimation for translation prediction. In *Proceedings of the Seventh Conference on Natural Language Learning at HLT-NAACL 2003 - Volume 4*, CONLL '03, pages 95–102, Stroudsburg, PA, USA. Association for Computational Linguistics.

Jesús González-Rubio, Daniel Ortiz-Martínez, and Francisco Casacuberta. 2010a. Balancing user effort and translation error in interactive machine translation via confidence measures. In *Proceedings of the ACL 2010 Conference Short Papers*, ACLShort '10, pages 173–177, Stroudsburg, PA, USA. Association for Computational Linguistics.

Jesús González-Rubio, Daniel Ortiz-Martínez, and Francisco Casacuberta. 2010b. On the use of confidence measures within an interactive-predictive machine translation system. In *Proceedings of 14th Annual Conference of the European Association for Machine Translation*.

Rebecca Knowles and Philipp Koehn. 2016. Neural interactive translation prediction. In *Proceedings of the Conference of the Association for Machine Translation in the Americas (AMTA)*.

Varvara Logacheva, Michal Lukasik, and Lucia Specia. 2016. Metrics for evaluation of word-level machine translation quality estimation. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 585–590, Berlin, Germany. Association for Computational Linguistics.

F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay. 2011. Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, 12:2825–2830.

Matss Rikters and Mark Fishel. 2017. Confidence Through Attention. In *Proceedings of the 16th Machine Translation Summit (MT Summit 2017)*, Nagoya, Japan.

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*, pages 65–68, Valencia, Spain. Association for Computational Linguistics.

Rico Sennrich, Barry Haddow, and Alexandra Birch. 2016. Edinburgh neural machine translation systems for WMT 16. In *Proceedings of the First Conference on Machine Translation (WMT)*.

Lucia Specia, Gustavo Paetzold, and Carolina Scarton. 2015. Multi-level translation quality prediction with quest++. In *ACL-IJCNLP 2015 System Demonstrations*, pages 115–120, Beijing, China.

Nicola Ueffing and Hermann Ney. 2005. Application of word-level confidence measures in interactive statistical machine translation. In *In Proceedings of EAMT 2005 (10th Annual Conference of the European Association for Machine Translation*, pages 262–270.

Joern Wuebker, Spence Green, John DeNero, Sasa Hasan, and Minh-Thang Luong. 2016. Models and inference for prefix-constrained machine translation. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 66–75, Berlin, Germany. Association for Computational Linguistics.