Evaluating Machine Translation Quality with Conformal Predictive Distributions

Patrizio Giovannotti

Department of Computer Science
Royal Holloway University of London

12th Symposium
on Conformal and Probabilistic Prediction with Applications

Machine Translation

Definition

The task of automatically translating text from one language to another using a computer program

Examples

- Google Translate, Yandex
- Embedded (Twitter, Amazon, Youtube etc.)

Machine Translation

Definition

The task of automatically translating text from one language to another using a computer program

Examples

- Google Translate, Yandex
- Embedded (Twitter, Amazon, Youtube etc.)

How do we assess quality?

- Is translation t good enough to be published or used?
- Is t better or worse than a t' generated by a competitor?
- Could we provide feedback about the translation quality?

Evaluating Quality in MT

- $s = (s_1, \ldots, s_m)$ sentence in the source language
- $t = (t_1, \ldots, t_n)$ the same sentence, translated by an MT system
- $\mathcal{R} = \{r_1, \ldots, r_{|\mathcal{R}|}\}$ set of reference translations

MT Evaluator

A function that accepts as input a tuple $\langle s, t, \mathcal{R} \rangle$ and outputs a quality score $\hat{q} \in \mathbb{R}$.

Evaluating Quality in MT – Examples

- Classic evaluator: a metric that quantifies the amount of **lexical overlap** between t and r_i. Examples:
  - BLEU (Bilingual Evaluation Understudy) score (2001)
  - METEOR (2005)
- The success of neural machine translation techniques inspired new metrics such as BERTScore (2020)
  - each word is encoded via a BERT model
  - $\hat{q}$ depends on the pairwise cosine similarity between words in t and r_i

Evaluating Quality in MT – Examples

- Classic evaluator: a metric that quantifies the amount of **lexical overlap** between t and r_i. Examples:
  - BLEU (Bilingual Evaluation Understudy) score (2001)
  - METEOR (2005)
- The success of neural machine translation techniques inspired new metrics such as BERTScore (2020)
  - each word is encoded via a BERT model
  - $\hat{q}$ depends on the pairwise cosine similarity between words in t and r_i
- Still unclear how well these metrics reflect a model's performance
- These metrics require a set $\mathcal{R}$ of reference translations.

Quality Estimation

- In many real-world situations, $\mathcal{R} = \emptyset$

Quality Estimation

- In many real-world situations, $\mathcal{R} = \emptyset$
- Transformers return the sum of log-probs of each word in the sentence
  - real number $c \in (-\infty, 0]$
  - can be used to rank different translation candidates
  - however it does not generally correlate with human judgement

Quality Estimation

- In many real-world situations, $\mathcal{R} = \emptyset$
- Transformers return the sum of log-probs of each word in the sentence
  - real number $c \in (-\infty, 0]$
  - can be used to rank different translation candidates
  - however it does not generally correlate with human judgement
- Datasets with human-annotated quality scores have been created
  - Direct assessment (DA)
  - Human translation error rate (HTER)
- Learn to compute quality scores $\hat{q}$ that approximate ground truth scores q^*
- Essentially a regression task

Datasets

Task 1 of the WMT 2020 conference

English	→	German	(En-De)
English	→	Chinese	(En-Zh)
Romanian	→	English	(Ro-En)
Estonian	→	English	(Et-En)
Sinhala	→	English	(Si-En)
Nepali	→	English	(Ne-En)

- Sentence pairs (s, t) labelled with q^*
- s extracted from Wikipedia and translated into t with a transformer-based NMT model trained on publicly available data
- Each pair (s, t) is manually labelled with a 0-100 score by a group of 3 independent annotators.

Uncertainty-Aware MT Evaluation

Only one existing approach, by Glushkova et al. (2021)

- Goal: predict distribution $\hat{P}_Q(q)$
- Use *dropout* or *deep ensembles* to obtain a set of quality scores $\mathcal{Q} = \{\hat{q}_1, \ldots, \hat{q}_N\}$
- Treat $\mathcal{Q}$ as a sample drawn from a Gaussian distribution $\Rightarrow$ estimate $\hat{\mu}$ and $\hat{\sigma}^2$
- Compute confidence intervals $I[q_{min}(\epsilon), q_{max}(\epsilon)]$ for $\epsilon \in [0, 1]$

Uncertainty-Aware MT Evaluation

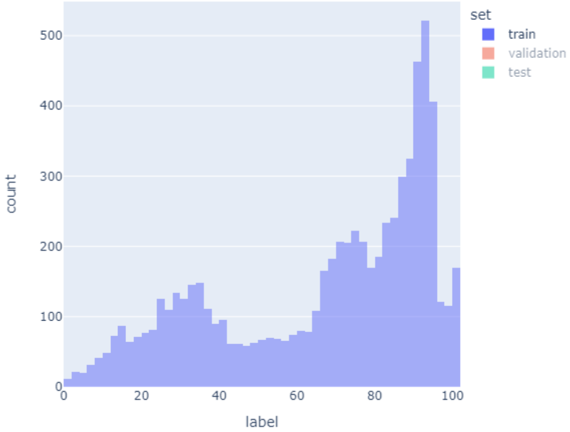
Only one existing approach, by Glushkova et al. (2021)

- Goal: predict distribution $\hat{P}_Q(q)$
- Use *dropout* or *deep ensembles* to obtain a set of quality scores $\mathcal{Q} = \{\hat{q}_1, \ldots, \hat{q}_N\}$
- Treat $\mathcal{Q}$ as a sample drawn from a Gaussian distribution $\Rightarrow$ estimate $\hat{\mu}$ and $\hat{\sigma}^2$
- Compute confidence intervals $I[q_{min}(\epsilon), q_{max}(\epsilon)]$ for $\epsilon \in [0, 1]$

Limitations

- Need to train several models or predict several times
- Need a further calibration step (no validity guarantees)
- Strong assumption about the shape of $\hat{P}_Q(q)$

Real-world distribution for Q



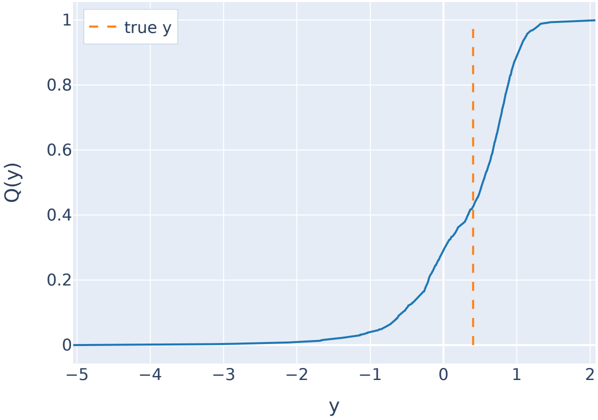
Label distribution for the Estonian → English training set

Conformal Predictive Distributions (CPD)

- Estimate the probability distribution of a continuous variable that depends on a number of features
- Assumption: data being generated independently by an unknown fixed distribution
- No prior required

→ CPDs provide probabilities that correspond to long-term frequencies (guaranteed coverage)

CPDs return a CDF



Conformal predictive distribution for a test example of the English → German dataset.
Values for the quality label y are normalized.

Implementation

- Inductive (split) CPD
- Package `crepes` (Boström, 2022)

Conformity Measure A

On a training set $z_1, \ldots, z_m$ of observations $z = (x, y)$:

$$A(z_1, \ldots, z_m, (x, y)) = \frac{y - \hat{y}}{\hat{\sigma}}$$

- $\hat{y}$ prediction for y
- $\hat{\sigma}$ estimate of the quality of $\hat{y}$

Underlying Algorithm

K-Nearest Neighbors regressor trained on two features:

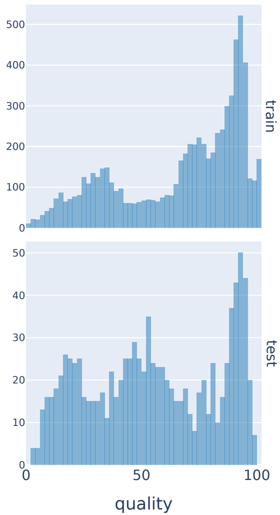
1. Quality $\hat{y}$ predicted by a `xlm-roberta-base` model, fine-tuned on our training set
2. Quality $\hat{y}'$ predicted by a second NMT model, included in our dataset

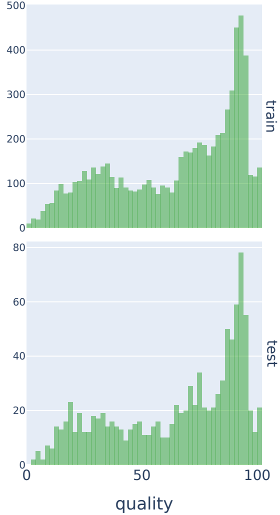
XLM-RoBERTa (Conneau et al., 2020)

- fine-tuned for 3 epochs
- keep the model achieving the best Pearson's correlation on eval set
- 3 different train/val/test splits

Importance of the Randomness Assumption

Estonian → English: before and after shuffling train / test splits





Evaluation Metrics

Expected Calibration Error (ECE)

$$\text{ECE} = \frac{1}{|\mathcal{E}|} \sum_{\epsilon \in \mathcal{E}} |\text{err}(\epsilon) - \epsilon|$$

where $\mathcal{E}$ is a set of significance levels $\epsilon \in [0, 1]$ and $\text{err}(\epsilon)$ is the error rate at a given significance level

Sharpness The degree of concentration of the predicted scores around the actual scores. In our case: average prediction interval width at 90% confidence

AUROC Area under the TPR-FPR curve (at different classification thresholds). We use it to assess the ability of our models to detect *critically wrong translations*, i.e. with quality q^* in the bottom decile of the test set.

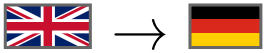
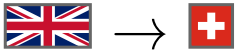
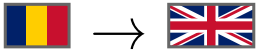
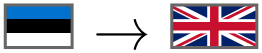
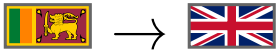
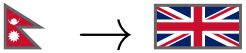
Baseline

- Map $\hat{q}$ to a Gaussian distribution $\mathcal{N}(q; \hat{\mu}, \hat{\sigma}^2)$ with

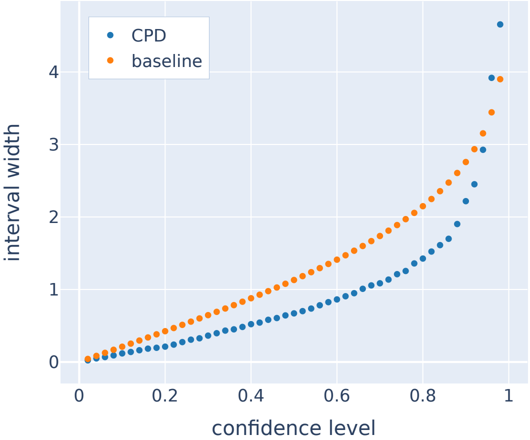
$$\hat{\mu} := \hat{q}, \;\; \hat{\sigma}^2 := \sigma_{\text{fixed}}$$

- σ_{fixed} obtained as average of the squared residuals $(\hat{q} - \hat{\mu})^2$ over the validation set
- Reasonable performance
- Fixed prediction intervals

Results

		%ECE	Sha@90%	AUC@10%
	baseline	13.88	2.81	0.63
	CPD	2.06	2.29	0.62
	baseline	3.19	2.51	0.78
	CPD	1.06	2.48	0.78
	baseline	3.96	1.92	0.92
	CPD	1.88	1.77	0.92
	baseline	4.20	2.45	0.83
	CPD	1.69	2.33	0.83
	baseline	2.82	2.61	0.80
	CPD	1.34	2.53	0.81
	baseline	3.11	2.40	0.79
	CPD	1.62	2.39	0.80

Sharpness



Sharpness on the English → German dataset. Each point is the prediction interval size averaged over all test examples for a particular confidence level $1 - \epsilon$.

Results (No Shuffling)

		%ECE	Sha@90%	AUC@10%
En-De	baseline	6.56	2.24	0.64
	CPD	3.75	1.99	0.64
En-Zh	baseline	1.31	2.21	0.73
	CPD	1.58	2.17	0.73
Ro-En	baseline	7.48	1.66	0.96
	CPD	4.04	1.54	0.96
Et-En	baseline	1.65	2.10	0.88
	CPD	2.78	2.02	0.88
Si-En	baseline	3.20	2.32	0.85
	CPD	4.03	2.31	0.85
Ne-En	baseline	5.23	2.04	0.88
	CPD	3.29	1.97	0.87

Future Work

- Explore methods for determining *if* and *when* the IID assumption has been violated during the training process
- *Retrain or not Retrain: Conformal Test Martingales for Change-point Detection* (Vovk et al., 2021)
- Conformal Prediction Under Distribution-Shift

Conclusion

- Novel approach to quality estimation for MT based on conformal predictive distributions
  - Generate a prediction interval which is larger the more is the uncertainty of the prediction
  - Predictions have guaranteed coverage under the IID assumption
- Allow for a range of useful *downstream* tasks
  - decide whether or not to publish a specific translation
  - to better rank translations produced by different MT models
  - to inform users about the confidence in the quality estimates
- Results confirm the importance of the IID assumption for the successful application of conformal methods in NLP tasks