Adaptive Quality Estimation for Machine Translation

Marco Turchi[1], Antonios Anastasopoulos[3], José G.C. de Souza[1,2], Matteo Negri[1]

[(1)] FBK – Fondazione Bruno Kessler, Trento, Italy, [(2)] University of Trento, Italy, [(3)] National Technical University of Athens, Greece

{turchi, desouza,negri}@fbk.eu, anastasopoulos.ant@gmail.com

Task: Machine Translation Quality Estimation (QE)

Given a (source, target) pair, predict the quality of the target *without reference translations.*

(One) application scenario: assess at run time the quality of MT suggestions in a Computer-assisted translation (CAT) environment.

Problem: adaptability

Since:
- The notion of MT output quality is highly subjective
- Each translation job has its own specificities

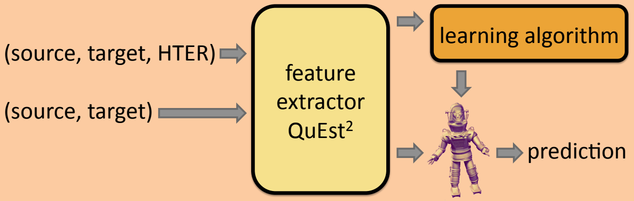
…QE components should be capable to *self-adapt* to:
- the behavior of specific users
- differences between training and test data

Solution: online learning

Idea:
Learn stepwise (either from scratch or by refining an existing model) *from user feedback*

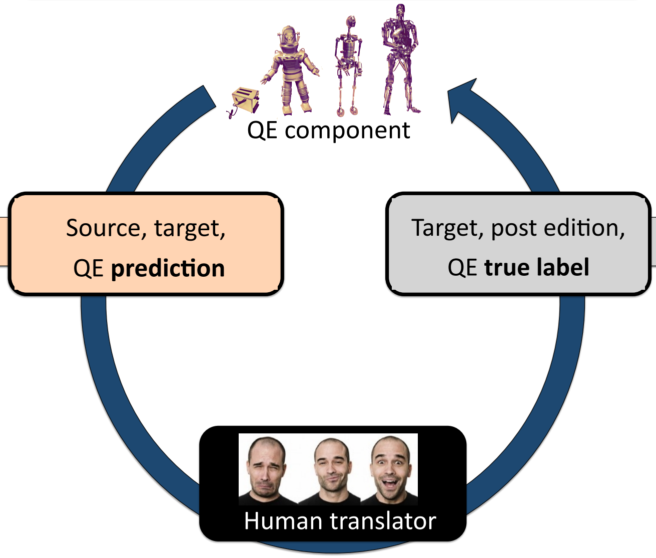
User = human translator
Feedback = distance between predicted labels and "true labels"
…calculated from MT post editions.

Predicting QE scores

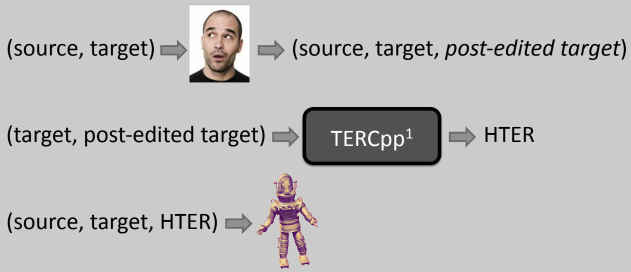


Features: 17 "baseline" QuEst features
Algorithms: SVR[3], OnlineSVR[4], Passive Aggressive Perceptron[5]

[2] QuEst - http://www.quest.dcs.shef.ac.uk
[3] Scikit - http://scikit-learn.org
[4] OnlineSVR - http://www2.imperial.ac.uk/~gmontana/onlinesvr.htm
[4] sofia-ml - https://code.google.com/p/sofia-ml



Collecting and exploiting user feedback



[1] TERCpp - http://sourceforge.net/projects/tercpp/

Experimental setup

Training/test data with *different label distributions*
- **WMT12 QE shared task – EN/ES** (artificial data partitions)
  - 1,832 training, 422 test sentences
- **MateCat data – EN/IT** (user and user+domain changes)
  - Legal (164 sentences) & Information Technology (280 sentences)
  - 8 professional translators

Comparison (Mean Absolute Error) between:
- **Adaptive**: built on top of an existing model
- **Empty**: only learns from the test set
- **Batch**: only learns from the training set
- **Baseline** (μ): label with the mean HTER calculated on training

Results on MateCat data (IT Vs Legal, Rad Vs Cons)

Train	Test	Δ HTER	μ MAE	Batch MAE	Adaptive MAE	Adaptive Alg	Empty MAE	Empty Alg
L cons	IT rad	24.5	26.4	27	18.2	OSVR	**16.6**	OSVR
IT rad	L cons	24.0	24.9	25.4	19.7	OSVR	**12.5**	OSVR
L rad	L cons	20.5	21.4	20.6	14.5	PA	**12.5**	OSVR
L cons	L rad	19.4	21.2	21.3	16.1	PA	**11.3**	OSVR
IT cons	L cons	13.5	17.3	17.5	15.7	OSVR	**12.5**	OSVR
IT cons	IT rad	12.8	19.2	19.8	17.5	OSVR	**16.6**	OSVR
L cons	IT cons	12.7	17.6	17.6	**15.1**	OSVR	15.5	OSVR
IT rad	IT cons	9.6	16.8	16.6	15.6	PA	**15.5**	OSVR
IT cons	L rad	8.3	12.3	13	**10.7**	OSVR	11.3	OSVR
L rad	IT rad	6.8	17	16.9	**16.2**	OSVR	16.6	OSVR
L rad	IT cons	5.0	15.4	16.2	**14.7**	OSVR	15.5	OSVR
IT rad	L rad	2.2	10.6	10.8	**10.5**	OSVR	11.3	OSVR

Take home messages

- Real-world scenarios raise new, interesting challenges for QE
  - Training/test data homogeneity, users' individual preferences, etc.
- Adaptability as a crucial capability (not only for CAT!)
  - Even in the same domain different user may show high ΔHTER
- Online learning from user corrections as a way to overcome the limitations of batch strategies
  - Use "empty" models (with OSVR) with highly heterogeneous data
- Use our open source tool!

http://hlt.fbk.eu/technologies/aqet