QuEst Integration in Okapi

FEISGILTT Dublin 2014

Yves Savourel

ENLASO Corporation

European Association for Machine Translation

This project is sponsored by the EAMT

This presentation was made possible by mylider

Project

- 6 months project sponsored by the **EAMT** (European Association for Machine Translation)
- **Gustavo Henrique Paetzold** = main developer doing most of the work
- **Lucia Specia** = project lead / coordinator
- Kashif Shah = helper on the QuEst side
- Yves Savourel = helper on the Okapi side

QuEst

- A translation quality estimation framework.
- Free, open source, cross-platform
- Use a set of extracted features from source and target to come up with a estimated score of the quality for a given segment.
- The prediction model is created from a set of bilingual training data annotated with human-estimated quality scores run through a learning algorithm (here SVM).

Okapi Framework

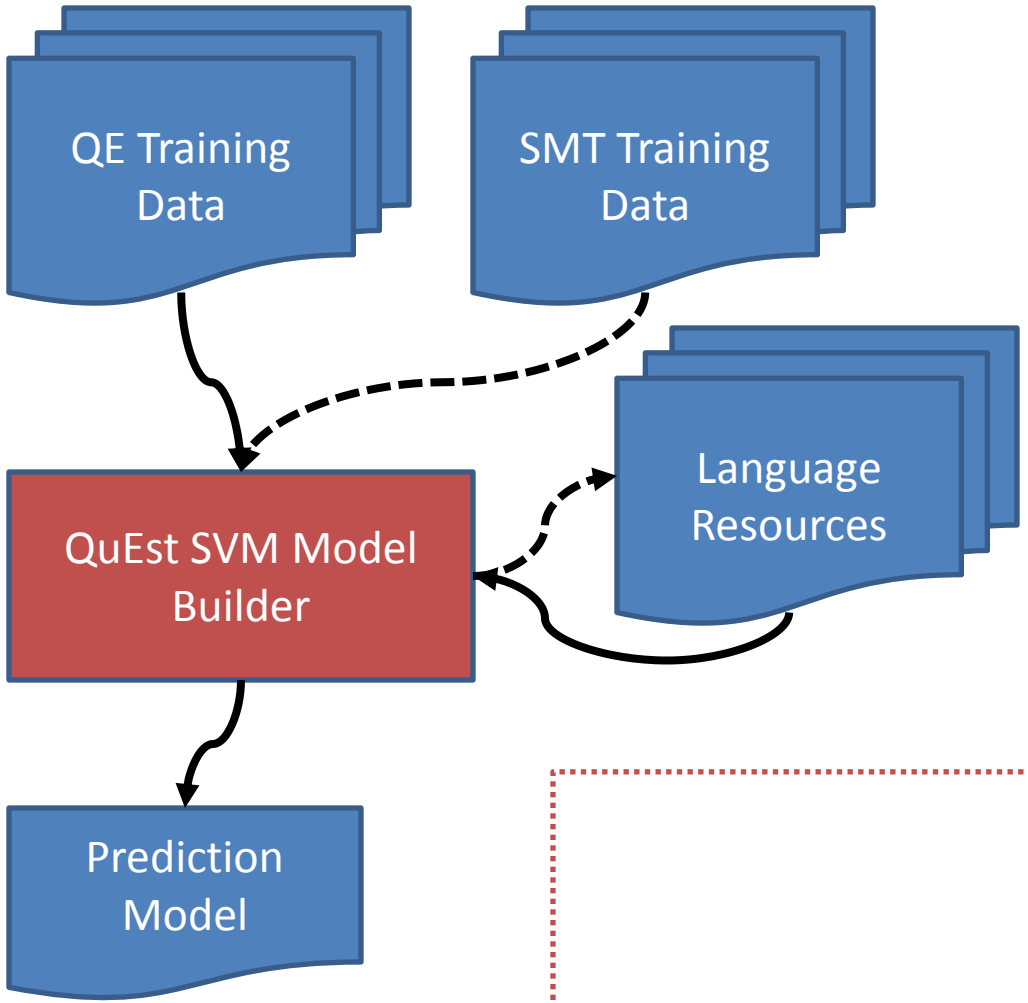
- Sets of components to build localization and translation tools and workflows.
- Free, open-source, cross-platform.
- Include MT connectors and components to create translation kits, so a component to generate translation quality estimation make sense.

Project objectives

1. Okapi components to annotate MT candidates with QuEst quality estimation scores.
2. Okapi component to create prediction models from sets of bilingual training data.
3. A pilot user study on ways to utilize the estimations in a workflow (e.g. filtering out under a given threshold, color display, etc.) Results published as a research paper.

QuEst SVM Model Builder

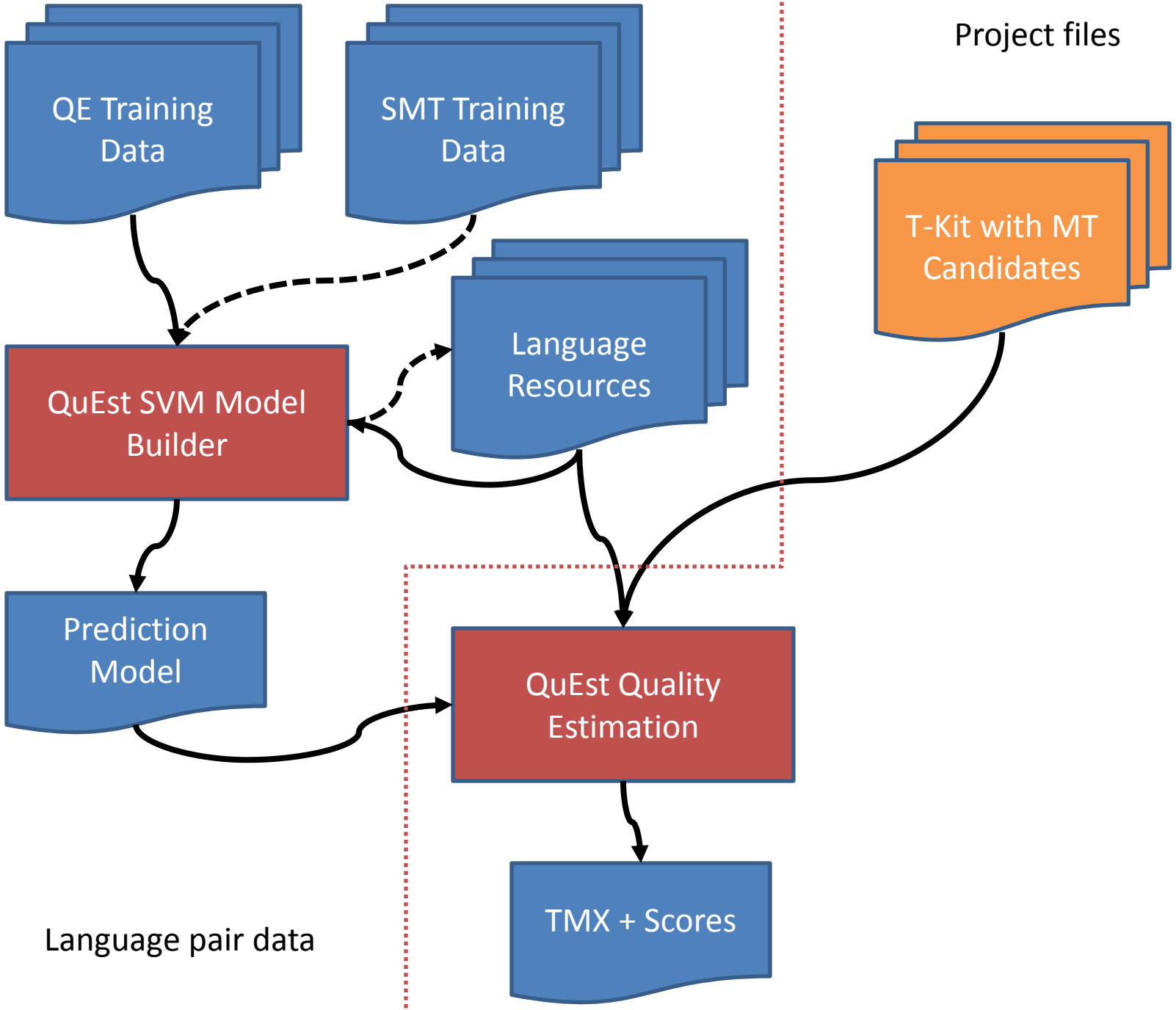
- Quality estimation training data input are:
  - Two sets of parallel monolingual files or one set of bilingual files (e.g. TMX, XLIFF, etc.)
  - File with one quality “label” (e.g. a value between 1 (bad) and 5 (good)) associated to each entry of the parallel text.
- Can use existing language resources or create the necessary ones from additional training data.
- Output: prediction model
  (and optionally: language resources)



Language pair data

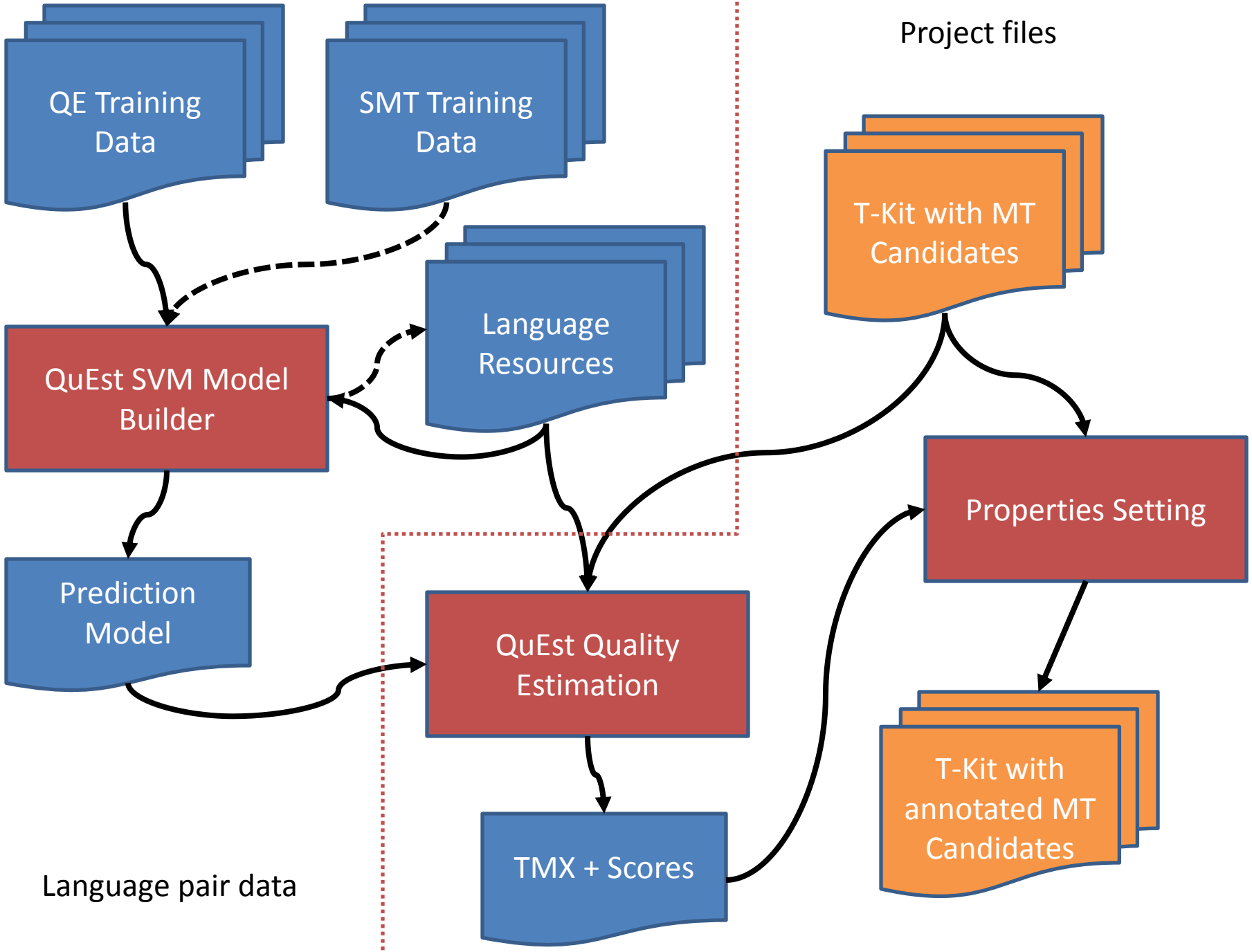
QuEst Quality Estimation

- Input documents are either:
  - T-Kit files (e.g. XLIFF) with MT candidates
  - Or two sets of parallel monolingual files
- Use existing prediction model and language resources to evaluate the translated segments and assign a score to each of them.
- Output: a TMX document with segments and scores (as properties).


Project files
QE Training Data
SMT Training Data
T-Kit with MT Candidates
Language Resources
QuEst SVM Model Builder
Prediction Model
QuEst Quality Estimation
TMX + Scores
Language pair data

Properties Setting

- Input documents:
  - T-Kit files (e.g. XLIFF) with MT candidates (same as in the previous step)
  - The TMX file with the segments + QuEst scores
- Output: annotated XLIFF documents (using ITS MT Confidence)


Project files
QE Training Data
SMT Training Data
T-Kit with MT Candidates
Language Resources
QuEst SVM Model Builder
Properties Setting
Prediction Model
QuEst Quality Estimation
T-Kit with annotated MT Candidates
TMX + Scores
Language pair data

Demonstration

- Part 1:
  - Extract a simple HTML document to XLIFF 1.2
  - Use Microsoft Translator Hub to get MT candidates
- Part 2 (= QuEst Quality Estimation):
  - Generate QuEst scores for the MT candidates and place them into a TMX document
- Part 3 (= Properties Setting):
  - Associate the QuEst scores to their translations in the XLIFF document (as ITS annotations)

A few links

- Okapi QuEst Project home:
  https://code.google.com/p/okapi-quest/
- Mailing list:
  https://groups.google.com/forum/#!forum/okapi-quest
- Download:
  http://okapi.opentag.com/snapshots/
  (the file `okapiQuEst-<version>.zip`)
- QuEst Project Home:
  http://www.quest.dcs.shef.ac.uk/