Multi-system machine translation using online APIs for English-Latvian

Matīss Rikters
University of Latvia
19 Raina Blvd.,
Riga, Latvia
matiss@lielakeda.lv

Abstract

This paper describes a hybrid machine translation (HMT) system that employs several online MT system application program interfaces (APIs) forming a Multi-System Machine Translation (MSMT) approach. The goal is to improve the automated translation of English – Latvian texts over each of the individual MT APIs. The selection of the best hypothesis translation is done by calculating the perplexity for each hypothesis. Experiment results show a slight improvement of BLEU score and WER (word error rate).

1 Introduction

MSMT is a subset of HMT where multiple MT systems are combined in a single system to complement each other's weaknesses in order to boost the accuracy level of the translations. Other types of HMT include modifying statistical MT (SMT) systems with rule-based MT (RBMT) generated output and generating rules for RBMT systems with the help of SMT [19].

MSMT involves usage of multiple MT systems in parallel and combining their output with the aim to produce better result as for each of the individual systems. It is a relatively new branch of MT and interest from researchers has emerged more widely during the last 10 years. And even now such systems mostly live as experiments in lab environments instead of real, live, functional MT systems. Since no single system can be perfect and different systems have different advantages over others, a good combination must lead towards better overall translations.

There are several recent experiments that use MSMT. Ahsan and Kolachina [1] describe a way of combining SMT and RBMT systems in multiple setups where each one had input from the SMT system added in a different phase of the RBMT system.

Barrault [3] describes a MT system combination method where he combines confusion networks of the best hypotheses from several MT systems into one lattice and uses a language model for decoding the lattice to generate the best hypothesis.

Mellebeek et al. [12] introduce a hybrid MT system that utilised online MT engines for MSMT. They introduce a system that at first attempts to split sentences into smaller parts for easier translation by the means of syntactic analysis, then translate each part with each individual MT system while also providing some context, and finally create the output from the best scored translations of each part (they use three heuristics for selecting the best translation).

Most of the research is done English – Hindi, Arabic – English and English – Spanish language pairs in their experiments. Where it concerns English - Latvian machine translation, no such experiments have been conducted.

This paper presents a first attempt in using an MSMT approach for the under-resourced English-Latvian language pair. Furthermore the first results of this hybrid system are analysed and compared with human evaluation. The experiments described use multiple combinations of outputs from two MT systems and one experiment uses three different MT systems.

Proceedings of the ACL 2015 Fourth Workshop on Hybrid Approaches to Translation (HyTra), pages 6–10,
Beijing, China, July 31, 2015.

2 System description

The main system consists of three major constituents – tokenization of the source text, the acquisition of a translation via online APIs and the selection of the best translation from the candidate hypotheses. A visualized workflow of the system is presented in Figure 1.

Currently the system uses three translation APIs (Google Translate[1], Bing Translator[2] and LetsMT[3]), but it is designed to be flexible and adding more translation APIs has been made simple. Also, it is initially set to translate from English into Latvian, but the source and target languages can also be changed to any language pair supported by the APIs.

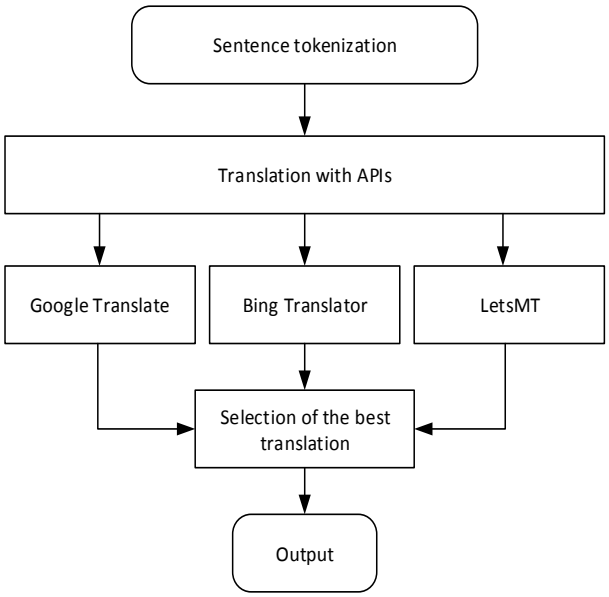

Figure 1: General workflow of the translation process

2.1 API description

Currently there are three online translation APIs included in the project – Google Translate, Bing Translator and LetsMT. These specific APIs were chosen for their public availability and descriptive documents as well as the wide range of languages that they offer. One of the main criteria when searching for translation APIs was the option to translate from English to Latvian.

2.2 Selection of the final translation

The selection of the best translation is done by calculating the perplexity of each hypothesis translation using KenLM [8]. First, a language model (LM) must be created using a preferably large set of training sentences. Then for each machine-translated sentence a perplexity score represents the probability of the specific sequence of words appearing in the training corpus used to create the LM. Sentence perplexity has been proven to correlate with human judgments close to the BLEU score and is a good evaluation method for MT without reference translations [7]. It has been also used in other previous attempts of MSMT to score output from different MT engines as mentioned by Callison-Burch et al. [4] and Akiba et al. [2].

KenLM calculates probabilities based on the observed entry with longest matching history w_f^n:

$$p(w_n \mid w_1^{n-1}) = p(w_n \mid w_f^{n-1}) \prod_{i=1}^{f-1} b(w_i^{n-1})$$

where the probability $p(w_n \mid w_f^{n-1})$ and backoff penalties $b(w_i^{n-1})$ are given by an already-estimated language model. Perplexity is then calculated using this probability:

$$b^{-\frac{1}{N}\sum_{i=1}^{N} \log_b q(x_i)}$$

where given an unknown probability distribution p and a proposed probability model q, it is evaluated by determining how well it predicts a separate test sample $x_1, x_2 \ldots x_N$ drawn from p.

3 System usage

The source code with working examples and sample data has been made open source and is available on GitHub[4]. To run the basic setup a Linux system is required with PHP and cURL installed. Before running, the user needs to edit the MSHT.php file and add his Google Translate, Bing Translator and LetsMT credentials as well as specify source and target languages (the defaults are set for English – Latvian).

The data required for an experiment is a source language text as a plain text file and a language model. The LM can be generated via KenLM using a large monolingual training corpus. The LM should be converted to binary format for more efficient usage.

[1] Google Translate API - https://cloud.google.com/translate/

[2] Bing Translator Control - http://www.bing.com/dev/en-us/translator

[3] LetsMT! Open Translation API - https://www.letsmt.eu/Integration.aspx

[4] Multi-System-Hybrid-Translator - https://github.com/M4t1ss/Multi-System-Hybrid-Translator

4 Experiments

The first experiments were conducted on the English – Latvian part of the JRC Acquis corpus version 2.2 [18] from which both the language model and the test data were retrieved. The test data contained 1581 randomly selected sentences. The language model was created using KenLM with order 5.

Translations were obtained from each API individually, combining each two APIs and lastly combining all three APIs. Thereby forming 7 different variants of translations. Google Translate and Bing Translator APIs were used with the default configuration and the LetsMT API used the configuration of TB2013 EN-LV v03[5].

Evaluation on each of the seven outputs was done with three scoring methods – BLEU [13], TER (translation edit rate) [16] and WER [9]. The resulting translations were inspected with a modified iBLEU tool [11] that allowed to determine which system from the hybrid setups was chosen to get the specific translation for each sentence.

The results of the first translation experiment are summarized in Table 2. Surprisingly all hybrid systems that include the LetsMT API produce lower results than the baseline LetsMT system. However the combination of Google Translate and Bing Translator shows improvements in BLEU score and WER compared to each of the baseline systems.

The table also shows the percentage of translations from each API for the hybrid systems. Although according to scores the LetsMT system was by far better than the other two, it seems that the language model was reluctant to favor its translations.

Since the systems themselves are more of a general domain and the first test was conducted on a legal domain corpus, a second experiment was conducted on a smaller data set containing 512 sentences of a general domain [15]. In this experiment only the BLEU score was calculated as it is shown in Table 1.

System	BLEU
Google Translate	24.73
Bing Translator	22.07
LetsMT	32.01
Hybrid Google + Bing	23.75
Hybrid Google + LetsMT	28.94
Hybrid LetsMT + Bing	27.44
Hybrid Google + Bing + LetsMT	26.74

Table 1: Second experiment results

System	BLEU	TER	WER	Translations selected			
				Google	Bing	LetsMT	Equal
Google Translate	16.92	47.68	58.55	100 %	-	-	-
Bing Translator	17.16	49.66	58.40	-	100 %	-	-
LetsMT	28.27	36.19	42.89	-	-	100 %	-
Hybrid Google + Bing	17.28	48.30	58.15	50.09 %	45.03 %	-	4.88 %
Hybrid Google + LetsMT	22.89	41.38	50.31	46.17 %	-	48.39 %	5.44 %
Hybrid LetsMT + Bing	22.83	42.92	50.62	-	45.35 %	49.84 %	4.81 %
Hybrid Google + Bing + LetsMT	21.08	44.12	52.99	28.93 %	34.31 %	33.98 %	2.78 %

Table 2: First experiment results

System	User 1	User 2	User 3	User 4	User 5	AVG user	Hybrid	BLEU
Bing	21,88%	53,13%	28,13%	25,00%	31,25%	31,88%	28,93%	16.92
Google	28,13%	25,00%	25,00%	28,13%	46,88%	30,63%	34,31%	17.16
LetsMT	50,00%	21,88%	46,88%	46,88%	21,88%	37,50%	33,98%	28.27

Table 3: Native speaker evaluation results

[5] https://www.letsmt.eu/TranslateText.aspx?id=smt-e3080087-866f-498b-977d-63ea391ba61e

5 Human evaluation

A random 2% (32 sentences) of the translations from the first experiment were given to five native Latvian speakers with an instruction to choose the best translation (just like the hybrid system should). The results are shown in Table 3. Comparing the evaluation results to the BLEU scores and the selections made by the hybrid MT a tendency towards the LetsMT translation can be observed among the user ratings and BLEU score that is not visible from the selection of the hybrid method.

6 Conclusion

This short paper described a machine translation system combination approach using public online MT system APIs. The main focus was to gather and utilize only the publically available APIs that support translation for the under-resourced English-Latvian language pair.

One of the test cases showed an improvement in BLEU score and WER over the best baseline.

In all hybrid systems that included the LetsMT API a decline in overall translation quality was observed. This can be explained by scale of the engines - the Bing and Google systems are more general, designed for many language pairs, whereas the MT system in LetsMT was specifically optimized for English – Latvian translations. This problem could potentially be resolved by creating a language model using a larger training corpus and a higher order for more precision.

7 Future work

The described system currently is only at the beginning of its lifecycle and further improvements are planned ahead. There are several methods that could improve the current system combination approach. One way is the application of other possible methods for selection of the best hypothesis.

For instance – the QuEst framework [17] can be used to extract various linguistic features for each sentence in the training corpora. Afterwards using the features along with a quality rating for each sentence a machine learning algorithm can train a model for predicting translation quality. The resulting model can then evaluate each candidate translation in a multi-system setup instead of perplexity.

Another path for hypothesis selection is the creation of a confusion network as described by Rosti, et al. [14]. This can be done with tools from either the Hidden Markov Toolkit[6] or the NIST Scoring Toolkit[7].

It would also be worth looking into any other forms of evaluating translations that do not require reference translations or MT quality estimation. For instance an evaluation using n-gram co-occurrence statistics as mentioned by Doddington [6] and Lin et al. [10] or quality estimation using tree kernels introduced by Cohn et al. [5].

Acknowledgements

This research work was supported by the research project “Optimization methods of large scale statistical models for innovative machine translation technologies”, project financed by The State Education Development Agency (Latvia) and European Regional Development Fund, contract No. 2013/0038/2DP/2.1.1.1.0/13/APIA/ VI-AA/029. The author would also like to thank Inguna Skadiņa for advices and contributions, and the anonymous reviewers for their comments and suggestions.

Reference

[1] Ahsan, A., and P. Kolachina. "Coupling Statistical Machine Translation with Rule-based Transfer and Generation, AMTA-The Ninth Conference of the Association for Machine Translation in the Americas." Denver, Colorado (2010).

[2] Akiba, Yasuhiro, Taro Watanabe, and Eiichiro Sumita. "Using language and translation models to select the best among outputs from multiple MT systems." Proceedings of the 19th international conference on Computational linguistics-Volume 1. Association for Computational Linguistics, 2002.

[3] Barrault, Loïc. "MANY: Open source machine translation system combination." The Prague Bulletin of Mathematical Linguistics 93 (2010): 147-155.

[4] Callison-Burch, Chris, and Raymond S. Flournoy. "A program for automatically selecting the best output from multiple machine translation

[6] HTK Speech Recognition Toolkit - http://htk.eng.cam.ac.uk/

[7] NIST Scoring Toolkit Version 0.1 - http://www1.icsi.berkeley.edu/Speech/docs/sctk-1.2/sctk.htm

engines." Proceedings of the Machine Translation Summit VIII. 2001.

[5] Cohn, Trevor, and Lucia Specia. "Modelling Annotator Bias with Multi-task Gaussian Processes: An Application to Machine Translation Quality Estimation." ACL (1). 2013.

[6] Doddington, George. "Automatic evaluation of machine translation quality using n-gram co-occurrence statistics." Proceedings of the second international conference on Human Language Technology Research. Morgan Kaufmann Publishers Inc., 2002.

[7] Gamon, Michael, Anthony Aue, and Martine Smets. "Sentence-level MT evaluation without reference translations: Beyond language modeling." Proceedings of EAMT. 2005.

[8] Heafield, Kenneth. "KenLM: Faster and smaller language model queries." Proceedings of the Sixth Workshop on Statistical Machine Translation. Association for Computational Linguistics, 2011.

[9] Klakow, Dietrich, and Jochen Peters. "Testing the correlation of word error rate and perplexity." Speech Communication 38.1 (2002): 19-28.

[10] Lin, Chin-Yew, and Eduard Hovy. "Automatic evaluation of summaries using n-gram co-occurrence statistics." Proceedings of the 2003 Conference of the North American Chapter of the Association for Computational Linguistics on Human Language Technology-Volume 1. Association for Computational Linguistics, 2003.

[11] Madnani, Nitin. "iBLEU: Interactively debugging and scoring statistical machine translation systems." Semantic Computing (ICSC), 2011 Fifth IEEE International Conference on. IEEE, 2011.

[12] Mellebeek, Bart, et al. "Multi-engine machine translation by recursive sentence decomposition." (2006).

[13] Papineni, Kishore, et al. "BLEU: a method for automatic evaluation of machine translation." Proceedings of the 40th annual meeting on association for computational linguistics. Association for Computational Linguistics, 2002.

[14] Rosti, Antti-Veikko I., et al. "Combining Outputs from Multiple Machine Translation Systems." HLT-NAACL. 2007.

[15] Skadiņa, Inguna, et al. "A Collection of Comparable Corpora for Under-resourced Languages." Proceedings of the Fourth International Conference Baltic HLT 2010. 2010.

[16] Snover, Matthew, et al. "A study of translation edit rate with targeted human annotation." Proceedings of association for machine translation in the Americas. 2006.

[17] Specia, Lucia, et al. "QuEst-A translation quality estimation framework." ACL (Conference System Demonstrations). 2013.

[18] Steinberger, Ralf, et al. "The JRC-Acquis: A multilingual aligned parallel corpus with 20+ languages." arXiv preprint cs/0609058 (2006).

[19] Thurmair, Gregor. "Comparingdifferentarchitecturesof hybridMachineTranslationsystems." (2009).