Multi-Task Minimum Error Rate Training for SMT

Patrick Simianer, Katharina Wäschle, Stefan Riezler

Department of Computational Linguistics
University of Heidelberg, Germany

Multi-Task Learning

- Multi-task learning aims at learning several different tasks simultaneously,
  - addressing **commonalities** through **shared parameters**
  - and modeling **differences** through **task-specific parameters**.
- Predestined application: Patent translation over classes of patents w.r.t. International Patent Classification (IPC)
  - **commonalities:** highly specialized legal jargon not found in everyday language, rigid textual structure including highly formulaic language.
  - **differences:** technological terminology specific to IPC class.

IPC Sections

- A Human Necessities
- B Performing Operations; Transporting
- C Chemistry; Metallurgy
- D Textiles; Paper
- E Fixed Constructions
- F Mechanical Engineering; Lighting; Heating; Weapons; Blasting
- G Physics
- H Electricity

Goal and Approach

Goal: Learn a translation system that performs well across several different patent sections, thus benefits from shared information, and yet is able to address the specifics of each patent section.

Approach: Machine learning approach to trading off optimality of parameter vectors for each task-specific model and closeness of these model parameters to average parameter vector across models.

Multi-Task Minimum Error Rate Training

- Assume specific setting: Not enough data for training generative SMT pipeline on all tasks, however, enough data for tuning for each specific task.
- In other words: How much gain is there in extending the standard tuning technique of minimum error rate training (MERT) to **multi-task MERT** for SMT.
- Also apply techniques for parameter averaging from distributed learning to a version of **averaged MERT**.

Parallel Patent Data

- MAREC: 19 million patent applications and granted patents, standardized format from four patent organizations (European Patent Office (EP), World Intellectual Property Organisation (WO), United States Patent and Trademark Office (US), Japan Patent Office (JP)), from 1976 to 2008.
- Extract bilingual abstract and claims sections from the EP and WO parts for German-to-English translation.
- Sentence splitting and tokenizing with Europarl tools[1].
- Sentence alignment with Gargantua 1.0b[2].

[1] http://www.statmt.org/europarl/

[2] http://sourceforge.net/projects/gargantua/

Distribution of IPC sections for de-en abstracts and claims

A	266,521	21.81%
B	384,517	31.47%
C	372,903	30.52%
D	50,579	4.14%
E	54,396	4.45%
F	149,370	12.22%
G	291,671	23.87%
H	228,147	18.67%

Parallel data for de-en patent translation

	train	dev	devtest	test
# parallel sents	1M	2K	2K	2K
avg. # tokens de	32,329,745	59,376	60,061	59,930
avg. # tokens en	36,005,763	69,584	70,700	70,331
year	1993-1995	2007	2008	2008

Multi-task learning objective

Objective: Minimize task-specific loss functions l_d under regularization of task-specific parameter vectors w_d towards an average parameter vector w_{avg}.

$$\min_{w_1,\ldots,w_D} \sum_{d=1}^{D} l_d(w_d) + \lambda \sum_{d=1}^{D} \|w_d - w_{\mathrm{avg}}\|_p^p \qquad (1)$$

Multi-task prediction

Prediction:

Task-specific weight vectors $w_d \in \{w_1, \ldots, w_D\}$ that have been adjusted to trade off task-specificity (small λ) and commonality (large λ).

or: Average weight vector w_{avg} as a global model.

Average MERT

$\texttt{AvgMERT}(w^{(0)}, D, \{c_d\}_{d=1}^{D})$:
for $d = 1, \ldots, D$ **parallel do**
  for $t = 1, \ldots, T$ **do**
    $w_d^{(t)} = \texttt{MERT}(w_d^{(t-1)}, c_d(w_d))$
  end for
end for
return $w_{\text{avg}} = \frac{1}{D} \sum_{d=1}^{D} w_d^{(T)}$

- Apply ideas from distributed learning (Zinkevich et al. NIPS'10) by basing the distribution strategy on task-specific partitions of data.

Multi-task MERT

regularization: Set p=1 in equation 1 to obtain an ℓ_1 regularizer.

clipping: Weight vector w_d is moved towards the average weight vector w_{avg} by adding or subtracting the penalty λ for each weight component $w_d[k]$, and clipped when it crosses the average.

code: Script wrapper around the `MERT` implementation of Bertoldi et al. 2009; licensed unter the LGPL; online at `http://www.cl.uni-heidelberg.de/statnlpgroup/mmert/`.

Multi-task MERT

$\mathtt{MMERT}(w^{(0)}, D, \{c_d\}_{d=1}^{D})$:
for $t = 1, \ldots, T$ **do**
  $w_{\text{avg}}^{(t)} = \frac{1}{D}\sum_{d=1}^{D} w_d^{(t-1)}$
  for $d = 1, \ldots, D$ **parallel do**
    $w_d^{(t)} = \mathtt{MERT}(w_d^{(t-1)}, c_d(w_d))$
    for $k = 1, \ldots, K$ **do**
      if $w[k]_d^{(t)} - w_{\text{avg}}^{(t)}[k] > 0$ **then**
        $w_d^{(t)}[k] = \max(w_{\text{avg}}^{(t)}[k], w_d^{(t)}[k] - \lambda)$
      else if $w_d^{(t)}[k] - w_{\text{avg}}^{(t)}[k] < 0$ **then**
        $w_d^{(t)}[k] = \min(w_{\text{avg}}^{(t)}[k], w_d^{(t)}[k] + \lambda)$
      end if
    end for
  end for
end for
return $w_1^{(T)}, \ldots, w_D^{(T)}, w_{\text{avg}}^{(T)}$

Experimental Setup

- Open-source Moses SMT system (Koehn et al. 2007); MERT implementation of Bertoldi et al. 2009.
- All systems use same phrase tables and language models, trained on 1M parallel data pooled from all IPC sections.
- *ind.* systems are tuned on each IPC section separately.
- *pooled* system is tuned on 2K sentences pooled from 250 sentences from each IPC section.
- `AvgMERT` and `MMERT` are algorithms described above.
- w_{avg} is global model produced as by-product in multi-task learning.

Experimental Evaluation

- All systems evaluated on 8 test sets, each consisting of 2K sentences from a separate IPC domain.
- Statistical significance of pairwise result differences assessed by p-values smaller than 0.05 using Approximate Randomization test (Riezler & Maxwell2005).
- statistically significant improvement over *ind* indicated by $*$
- statistically significant improvement over *pooled* indicated by $+$
- statistically significant improvement over `AvgMERT` indicated by $\#$

Experimental Results

section	*ind.*	*pooled*	AvgMERT	MMERT	w_{avg}
A	0.5187	0.5199	**0.5213***	0.5195$^{\#}$	0.5196$^{\#}$
B	0.4877	0.4885	0.4908^{*+}	0.4911*	**0.4921**$^{*\#}$
C	0.5214	0.5175	0.5199^{*+}	**0.5218**$^{\#}$	0.5162$^{*\#}$
D	0.4724	0.4730	0.4733	**0.4736**	0.4734
E	0.4666	0.4661	0.4679^{*+}	0.4669	**0.4685***
F	0.4794	0.4801	0.4811*	0.4821*	**0.4830**$^{*\#}$
G	0.4596	0.4576	0.4607^{+}	0.4606	**0.4610***
H	0.4573	0.4560	0.4578	**0.4581**	**0.4581**

Discussion

- *pooled* shows no s.s. improvement over *ind.*
- Best results (**bold face**) achieved by `AvgMERT`, `MMERT`, or w_{avg}.
- Best results are small, but statistically significant improvements over *ind.* and *pooled*.
- Significant degradation on section C ("chemistry") by averaging techniques due to expeptional character of chemical formulae and compound names.
- Interpretation of small improvements with a grain of salt, however, hope for larger improvments with larger feature sets.