Learning Bilingual Projections of Embeddings for Vocabulary Expansion in Machine Translation

The University Of Sheffield.
UNIVERSITÄT DES SAARLANDES
German Research Center for Artificial Intelligence

Pranava Madhyastha
&
Cristina España-Bonet

— Motivation —

- Problem: Unseen words in parallel corpora cannot be translated
- Proposal: Exploit monolingual corpora by:
  1. learn word embeddings on source and target data,
  2. map the two spaces using a word–word dictionary,
  3. integrate the pairs in a translation system

— The Probabilistic Model —

- SMT, log-linear model

$$\Pr(t|s) \sim \exp\left\{\sum \lambda_m h_m(s,t)\right\}$$

- BWE, bilinear model

$$\Pr(t|s) \sim \exp\left\{\phi(s)^\top W \phi(t)\right\}$$

— Log-Bilinear Model —

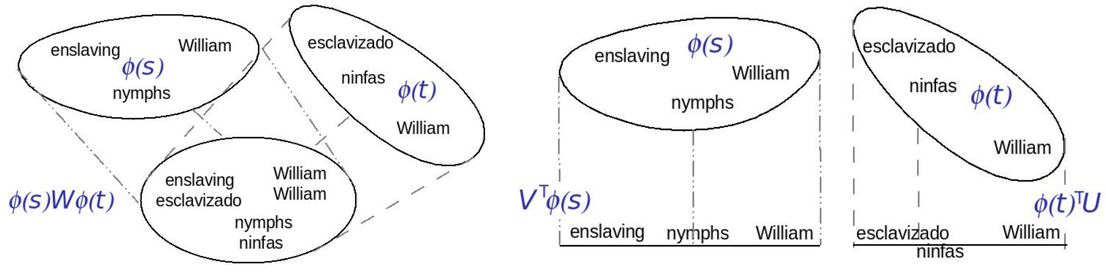
$$\Pr(t|s;W) = \frac{\exp\left\{\phi(s)^\top W \phi(t)\right\}}{\sum_{t'} \exp\left\{\phi(s)^\top W \phi(t)\right\}}$$

- Maximise negative log-likelihood with nuclear norm regularisation

$$-\sum_{t,s} \log \Pr(t|s;W) + \lambda \|W\|_*$$

— Compression Using Low-Rank Penalties —

$$\underbrace{[t_1\, t_2 \cdots\cdots t_n] \begin{bmatrix} u_{11} & \cdots & u_{1k} \\ u_{21} & \cdots & u_{2k} \\ \vdots & \vdots & \vdots \\ \vdots & \vdots & \vdots \\ u_{n1} & \cdots & u_{nn} \end{bmatrix}}_{\phi(t)^\top U} \underbrace{\begin{bmatrix} \sigma_1 & \cdots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \cdots & \sigma_k \end{bmatrix}}_{\Sigma} \underbrace{\begin{bmatrix} v_{11} & \cdots\cdots & v_{1n} \\ \vdots & \vdots & \vdots \\ v_{k1} & \cdots\cdots & v_{kn} \end{bmatrix} \begin{bmatrix} s_1 \\ s_2 \\ \vdots \\ \vdots \\ s_n \end{bmatrix}}_{V^\top \phi(s)}$$



- Things we can do (CWs)

s:	galaxy	nymphs
t:	galaxia	ninfas
t:	planeta	ninfa
t:	galaxias	dioses
...		...

- Things we cannot do (NEs, MWEs)

s:	Stuart	folksong
t:	William	música
t:	Henry	folclore
t:	Thomas	folklore
t:	Estuardo (#48)	canción

— I. Dictionary Induction —

- Setting: We follow Upadhyay *et al.* (2016)
- Top-10 accuracy (%) for English–German and English–French:

		Methods with Soft Supervision			
l_1	l_2	BiCCA	BiVCD	Ours-300	Ours-100
en	de	72.4	62.5	73.8	71.1
en	fr	70.1	68.8	72.1	69.7

Upadhyay *et al.*, "Cross-lingual Models of Word Embeddings: An Empirical Comparison", ACL, 2016.

— Observations —

- Without compression, our method performs significantly better than BiVCD and slightly better than BiCCA
- We can compress 2/3 without a significant loss in performance
- Strong supervised methods such as BiSkip are better (accuracy ~ 79%)

— Experimental Setting II —

Monolingual corpora: Wikipedia + Quest
Number of words: $2.3 \cdot 10^9$ en; $0.8 \cdot 10^9$ es
Embeddings: $2.0 \cdot 10^6$ en; $0.8 \cdot 10^6$ es (word2vec, cbow, 300D)
Dictionary: Apertium bilingual dict
Number of words: 34,806 (train+val.)

	Tokens	OOV_{all}	OOV_{CW}
NewsTest	64810	1590 (2.5%)	296 (0.5%)
WikiTest	11069	798 (7.2%)	201 (1.8%)

- Notice: low number of OOV content words (CW)!

— II. MT Automatic Evaluation —

	NewsTest			WikiTest		
	TER	BLEU	MTR	TER	BLEU	MTR
copyOOV	57.9	22.9	47.1	58.5	21.9	45.8
BWE_{all50}	58.3	22.2	45.8	58.4	21.9	44.8
BWE_{CW50}	57.7	23.1	47.1	56.2	24.2	48.5
BWE_{CW10}	57.8	23.1	47.1	55.6	24.7	49.1
BLM	55.4	25.8	49.2	52.6	30.6	51.0
BLM+BWE_{all50}	55.9	24.9	47.8	51.0	32.2	52.1
BLM+BWE_{CW50}	55.5	25.6	49.0	49.5	33.9	54.9
BLM+BWE_{CW10}	55.3	25.9	49.0	49.1	34.6	55.5

BLM: Big Language Model; CWn: top-n Content Words

— Manual Evaluation (BLM+BWE_{CW50}, WikiTest) —

- Accuracy in validation: ~ 82% (on the dictionary)
- Accuracy@50: 68%
- OOVs translated correctly: 22%
- Absolute numbers: 45 OOVs/11069 tokens improved
  ⇒ Huge impact on MT!

— Observations —

- Neighbouring words change with the choice of the OOV translation, that is a cause for the large improvement
- BWE augments and supports LM
- Our BLM+BWE_{CW10} gives 4 BLEU points performance boost

— Conclusions —

- We estimate bilingual word embeddings and, as a by-product, we compress the initial ones
- The addition of our new translation options to a mere 1.8% of the words leads to a significant relative improvement of a 13% in BLEU
- Next step: resolve multiword expressions using a similar model