Latent Domain Phrase-based Models for Adaptation

Hoang Cuong and **Khalil Sima'an**

ILLC, University of Amsterdam

UNIVERSITEIT VAN AMSTERDAM

Acknowledgement: **Ivan Titov**, **Gideon Wenniger**.

EMNLP 2014, Doha, Qatar.

Phrase-based models with Mix-of-Domains Data

We have **Big DATA** to train large-scale phrase-based systems.

- Thanks to Europarl, UN, Common Crawl, ...

Wait ...

- Data come from **very different domains**.
- **How does this affect phrase-based models performance**?

Bigger data $\neq$ better system performance

Exemplifying the mix-of-domains corpora effect

Experiment:

- **Train** phrase-based systems on mix-of-domain corpora, including *EuroParl*, *Common Crawl*, *UN*, *News Commentary*, *Software*, *Hardware*, etc.
  - Learning very large phrase-tables, with roughly **100M**, **200M**, and **300M** entries.
- **Test** on a specific test set related to **Electronics**.

Mixed data = 1M training sentence pairs

Input: *Se puede crear un archivo autodescodificable cuando el archivo codificado se abre con la contraseña maestra.*

Reference: *A self-decrypting file can be created when the encrypted* **file** *is* **opened with** *the* **master password**.

Output: *To create an file autodescodificable when the* **file** *codified* **commenced with** *the* **password teacher**.

Why? Mixed data = relevant + irrelevant translations!

maestra → **master** (**computer**);
maestra → **teacher** (**education**); maestra → **dean** (**education**);
maestra → **crack** (**other**), maestra → ...

Mixed data = 2M training sentence pairs

Input: *El reproductor puede reproducir señales de audio grabadas en mix-mode cd, cd-g, cd-extra y cd text.*

Reference: *The player* **can play back audio signals** *recorded in mix-mode cd, cd-g, cd-extra and cd text.*

Output: *The player* **can reproduce signs of audio** *recorded in mix-mode cd, cd-g, cd-extra and cd text.*

Why? Mixed data = relevant + irrelevant translations!

reproducir → **reproduce** (**science**);
reproducir → **play back** (**electronics**);
reproducir → **render** (**software**);
reproducir → ...

Mixed data = 4M training sentence pairs

Input: *Repite todas las pistas (únicamente cds de vídeo sin pbc)*

Reference: *Repeat* **all tracks** *(non-pbc video cds only)*

Output: *Repeated* **all avenues** *(only cds video without pbc)*

Why? Mixed data = relevant + irrelevant translations!

pistas → **tracks** (**electronics**);
pistas → **runway** (**news**);
pistas → **avenues** (**wrong alignment**);
pistas → **arena** (**electronics**);
pistas → ...

The Challenge

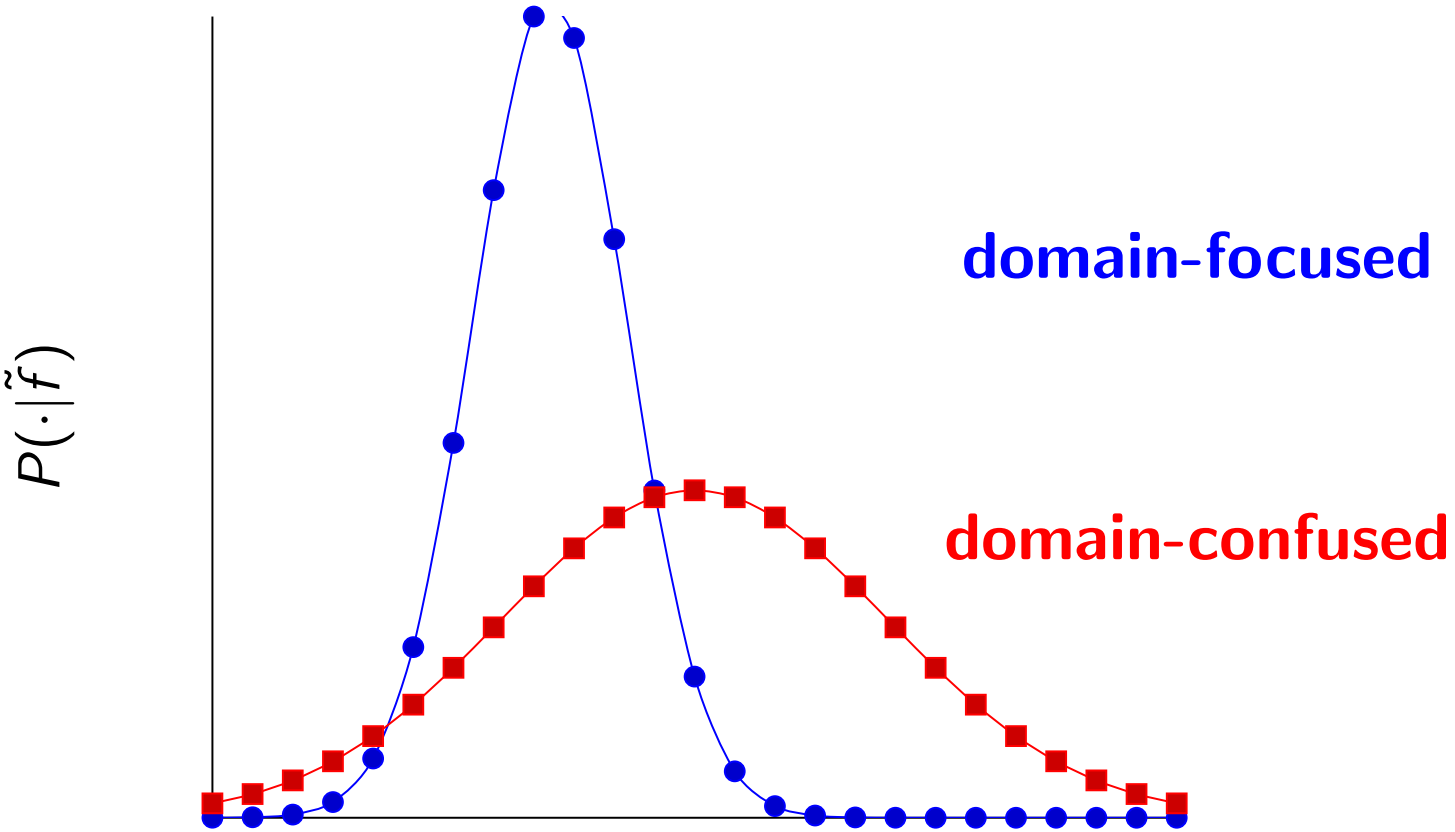
Problem: What we induced is the **domain-confused** statistics!

- The statistics that reflects translation preferences **averaged** over diverse domains.

Challenge: How to learn **domain-focused** statistics?

- The statistics that is induced with respect to the **target domain**, using in-domain data as **priors**.

Domain-confused vs. domain-focused statistics



Translation options over a phrase $\tilde{f}$

We view this as **in-domain corpus focused training**

Our Approach

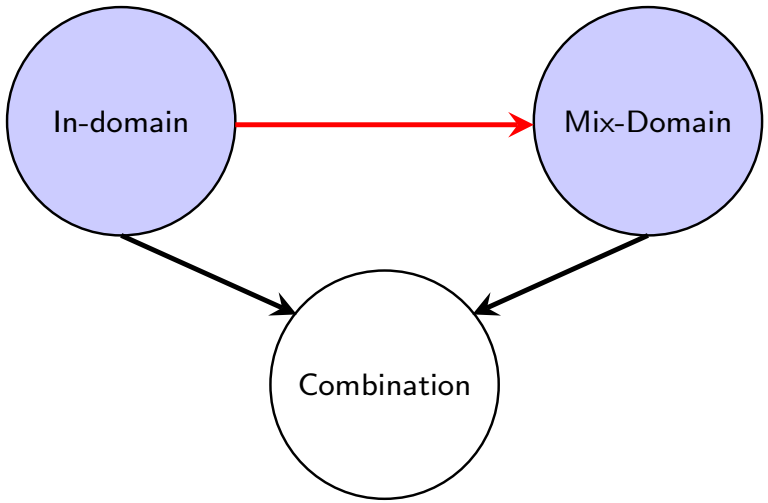
Materials

- Introduce a **latent domain variable**, D to represent domains of data.
  - D_1 - in-domain
  - D_0 - irrelevant domain.

Our Goal

- Learn the statistics with respect to (w.r.t) latent Ds.
  - **Phrase translation stats w.r.t.** D**s**,
    - $P(\tilde{e}|\ \tilde{f}, D)$ for each phrase pair $\langle \tilde{e},\ \tilde{f} \rangle$.
  - Lexical weight stats w.r.t. Ds.
  - Reordering stats w.r.t. Ds.

Our Approach



Complementary contribution in DA literature

- The statistics can be used as **additional** features, improving mix-of-domain systems.
- Nonetheless, this work prefers experiments that **replace** the domain-confused statistics (**more fun**).

Latent Domain Phrase-based Models

How to model $P(\tilde{e}|\ \tilde{f}, D)$?

- Be careful to play with **Expectation Maximization** at phrase level!

We found that a simple decomposition really works!

$$P(\tilde{e}|\ \tilde{f}, D) \propto \overbrace{P(\tilde{e}|\ \tilde{f})}^{\textit{Relative Frequency}} \quad \overbrace{P(D|\ \tilde{e},\ \tilde{f})}^{\textit{Expectation Maximization}} .$$

- $P(D|\ \tilde{e},\ \tilde{f})$: phrase-relevance models
  - $P(D_1|\ \tilde{e},\ \tilde{f})$ models how *relevant* a phrase pair
  - $P(D_0|\ \tilde{e},\ \tilde{f})$ models how *irrelevant* a phrase pair

Latent Domain Phrase-based Models

$$P(\tilde{e}|\ \tilde{f}, D) \propto \overbrace{P(\tilde{e}|\ \tilde{f})}^{\textit{Relative Frequency}} \quad \overbrace{P(D|\ \tilde{e},\ \tilde{f})}^{\textit{Expectation Maximization}} .$$

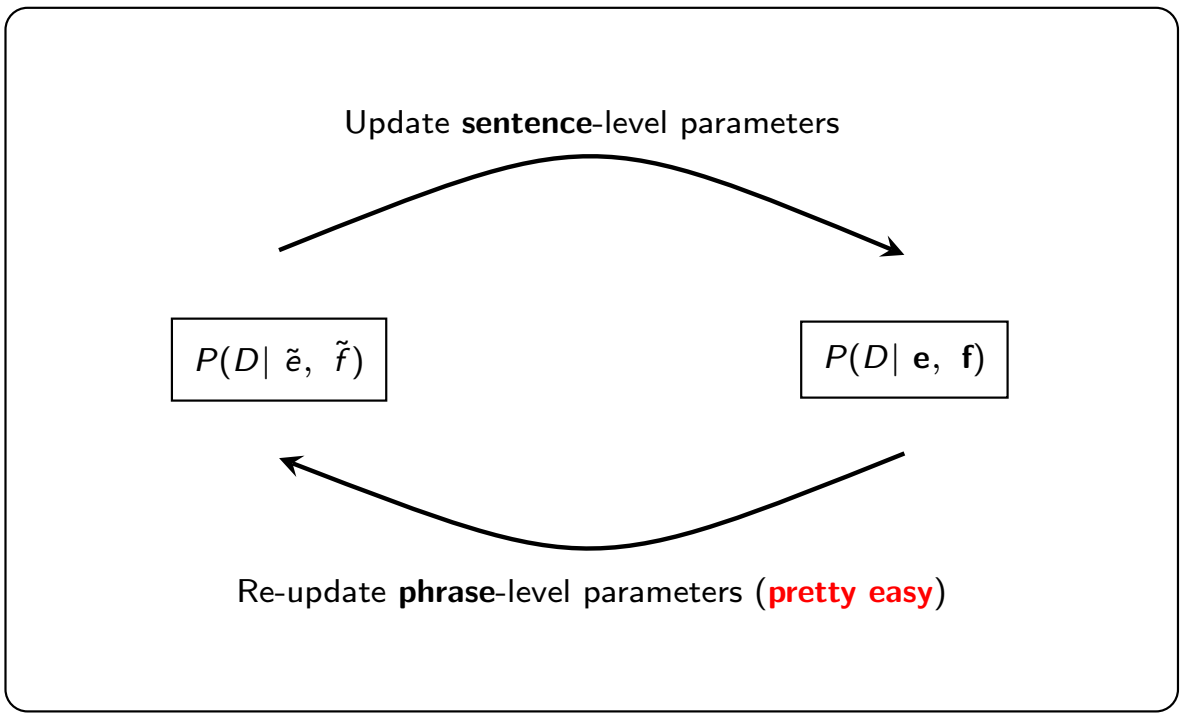
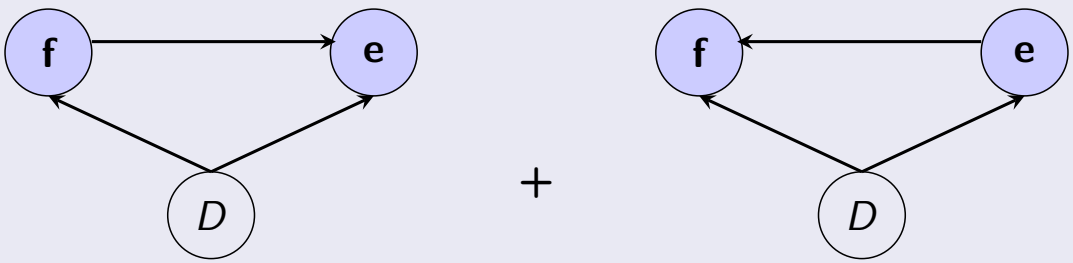


Figure: Reminiscent of EM learning framework.

Sentence-relevance models

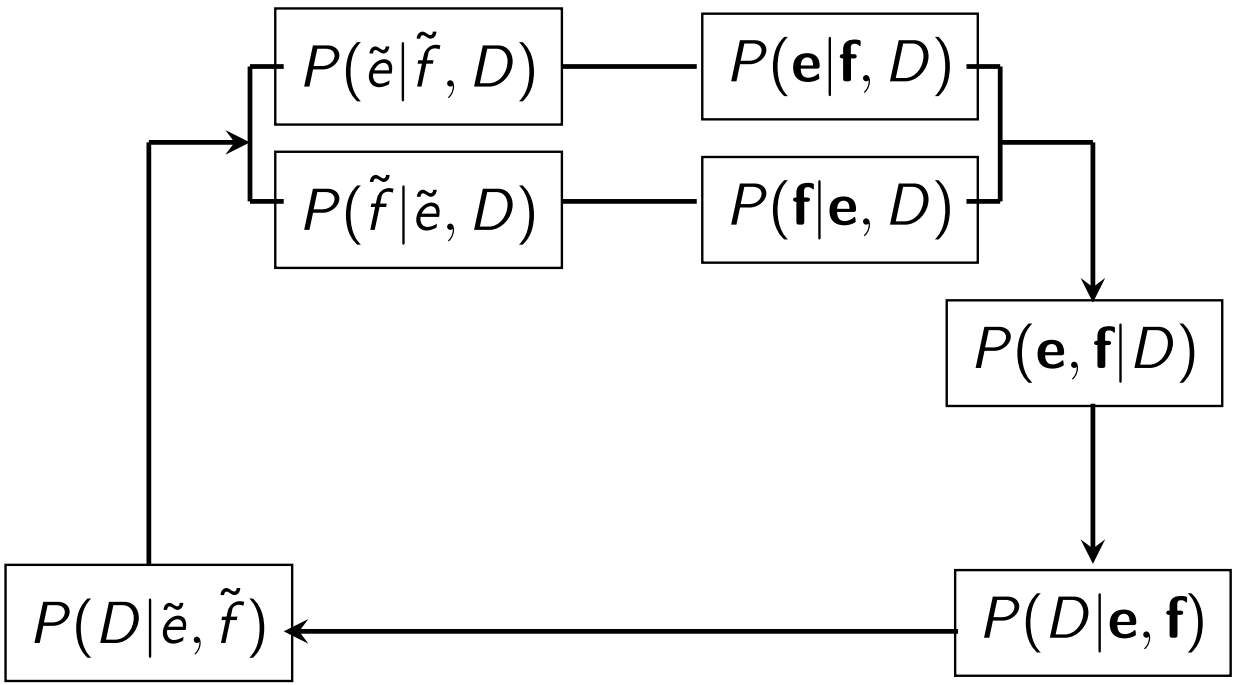
$$P(D|\,\mathbf{e},\,\mathbf{f}) \propto \overbrace{P(\mathbf{e},\,\mathbf{f}|\,D)}^{\textbf{generative process}} \quad \overbrace{P(D)}^{prior(easy)}$$



$$\overbrace{P(\mathbf{e}|\,D)}^{easy\ (and\ minor)}\ ,\quad \overbrace{P(\mathbf{f}|\,D)}^{easy\ (and\ minor)}\ ,\ \overbrace{P(\mathbf{f}|\,\mathbf{e},D)}^{hard},\ \overbrace{P(\mathbf{e}|\,\mathbf{f},D)}^{hard}$$

$$P(\mathbf{f}|\,\mathbf{e},D) := \prod_{\langle\tilde{e},\tilde{f}\rangle} P(\tilde{f}|\,\tilde{e},D); P(\mathbf{e}|\,\mathbf{f},D) := \prod_{\langle\tilde{e},\tilde{f}\rangle} P(\tilde{e}|\,\tilde{f},D)$$

Learning Framework



How do we start the training procedure?

Train $P(\tilde{e}|\ \tilde{f})$ on in-domain data, assuming it is $P(\tilde{e}|\ \tilde{f},\ D_1)$.
Train $P(\tilde{e}|\ \tilde{f})$ on mix-domain data, assuming it is $P(\tilde{e}|\ \tilde{f},\ D_0)$.

Experiments

We are **particularly** interested in two questions:

- How much translation improvement yielded by $P(\tilde{f}|\ \tilde{e}, D_1)$ over $P(\tilde{f}|\ \tilde{e})$?
- How many iterations needed to train the framework?

Other results can be found in the paper:

- Improvement yielded by adapting **lexical weights**
- Improvement yielded by adapting **reordering models**
- Experimental results on **various adaptation tasks**
- How our research **complementary contributes** to a full DA system

Data Preparation

- Mixed corpora of $1M$, $2M$, $4M$ sentences
  - EuroParl, Common Crawl, UN, News Commentary, Software, Hardware, and Pharmacy.
- An in-domain data set:
  - $109K$ in-domain sentence pairs.
  - Domain: TAUS **Electronics**.[1]

[1] Thanks to Translation Automation Society (TAUS.com)

Results

Data	System	Avg	Δ	p-value
1M	Domain-*confused* stats	19.91	–	–
	Domain-*focused* stats	20.64	**+0.73**	**0.0001**
2M	Domain-*confused* stats	20.54	–	–
	Domain-*focused* stats	21.41	**+0.87**	**0.0001**
4M	Domain-*confused* stats	21.44	–	–
	Domain-*focused* stats	22.62	**+1.18**	**0.0001**

Table: Translation Improvements.

Translation Examples

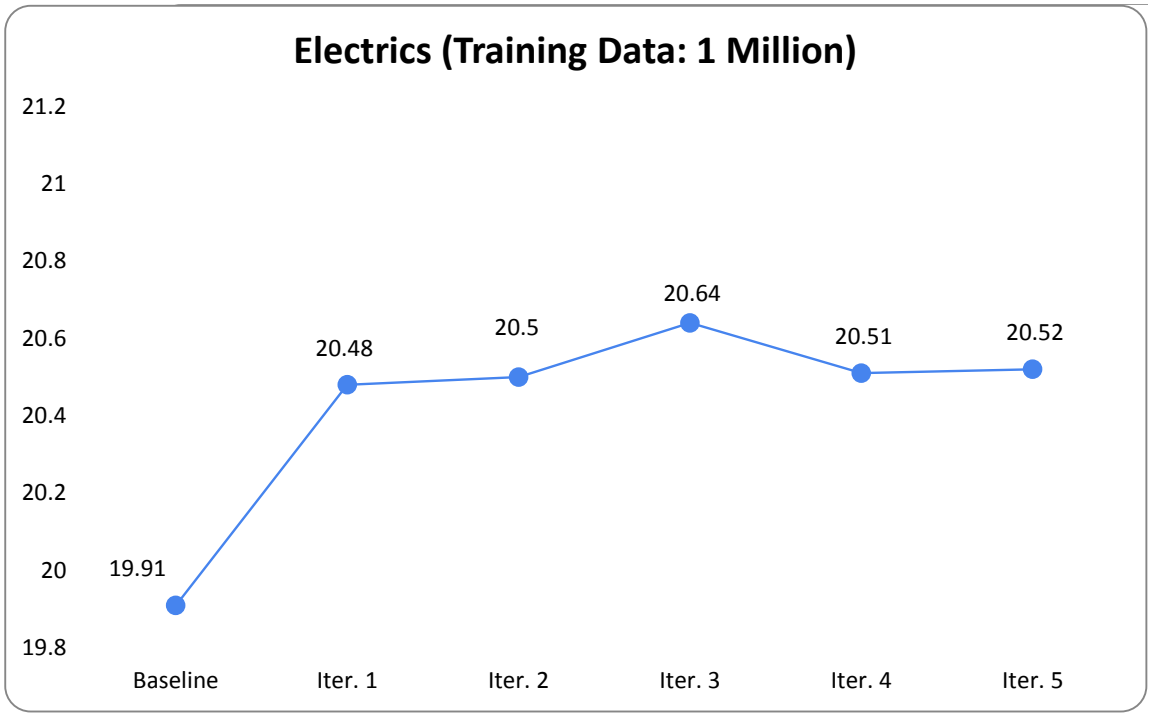
Input *El reproductor puede* **reproducir señales** *de audio grabadas en mix-mode cd, cd-g, cd-extra y cd text.*

Baseline *The player can* **reproduce signs of audio** *recorded in mix-mode cd, cd-g, cd-extra and cd text.*

Our System *The player* **can play signals audio** *recorded in mix-mode cd, cd-g, cd-extra and cd text.*

	señales		**reproducir**	
Entries	**signals**	**signs**	**play**	**reproduce**
Domain-*confused* stats	0.29	0.36	0.15	0.20
Domain-*focused* stats	0.37	0.17	0.34	0.16

Training iterations and averaged entropies



Baseline	Iter. 1	Iter. 2	Iter. 3	Iter. 4	Iter. 5
0.210	0.187	0.186	0.185	0.185	0.184

Table: Average entropy of distributions.

Final Discussions - Why is derived domain-focused statistics better?

$$P(\tilde{e}|\ \tilde{f}, D) \propto \overbrace{P(\tilde{e}|\ \tilde{f})}^{\textit{Relative Frequency}} \quad \overbrace{P(D|\ \tilde{e},\ \tilde{f})}^{\textit{Expectation Maximization}} .$$

- When $P(D|\ \tilde{e}, \tilde{f})$ estimation is **reliable**, $P(\tilde{e}|\ \tilde{f}, D)$ is really **meaningful**!
- When $P(D|\ \tilde{e}, \tilde{f})$ estimation is **NOT reliable**, i.e., phrases are **rare**,
  - $P(\tilde{e}|\ \tilde{f}, D) \rightarrow 1.0$
  - $P(\tilde{e}|\ \tilde{f}) \rightarrow 1.0$.
  - $P(\tilde{e}|\ \tilde{f}, D)$ and $P(\tilde{e}|\ \tilde{f})$ are equally "informative".

Final Discussions - Challenges?

- EM maximizes the **likelihood**, but **no tight correlation** between the likelihood and **translation performance**.
  - EM **sharpens** $P(D|\ \tilde{e},\ \tilde{f})$ without taking care whether it is harmful or not!

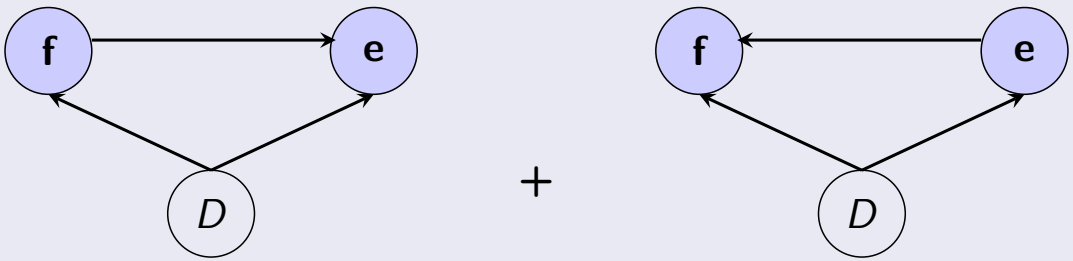
	señales	
Entries	**signals**	**signs**
Baseline	0.29	0.36
Iter. 1	0.36	0.23
Iter. 2	0.37	0.19
Iter. 3	0.37	0.17
Iter. 4	0.37	0.16
Iter. 5	0.37	0.15

Final Discussions - Challenges?

- In some cases, though not so often, performance could significantly drop off at Iteration 2!
- How to avoid the overfitting?
  - We hypothesize that back-off models can help!

Final Discussions - Challenges?

$$P(D|\ \mathbf{e},\ \mathbf{f}) \propto \overbrace{P(\mathbf{e},\ \mathbf{f}|\ D)}^{\textbf{generative process}}\ P(D)$$



$$\overbrace{P(\mathbf{f}|\ \mathbf{e}, D)}^{hard}, \overbrace{P(\mathbf{e}|\ \mathbf{f}, D)}^{hard}$$

Naive translation models:

$$P(\mathbf{f}|\ \mathbf{e}, D) := \prod_{\langle \tilde{e}, \tilde{f} \rangle} P(\tilde{f}|\ \tilde{e}, D); P(\mathbf{e}|\ \mathbf{f}, D) := \prod_{\langle \tilde{e}, \tilde{f} \rangle} P(\tilde{e}|\ \tilde{f}, D)$$

Final Discussions - Challenges?

- How to **generalize** 2 domains to N domains (N is arbitrary)?
  - I personally tried a lot, but did not achieve so much improvement!

Thank You!

Bibliography I