Domain Adaptation for Statistical Machine Translation

Amittai Axelrod

University of Maryland

MT Marathon in the Americas
2015

Domain??

- Not a well-defined concept.
- Should be based on some notion of textual similarity:
  - Lexical choice
  - Grammar
  - Topic
  - Style
  - Genre
  - Register
  - Intent

Domain Adaptation

- Domain = particular contextual setting.
  Here we use “domain” to mean “corpus”.

- Some domains have little available bilingual data:
  - Swiss Alpine route guides from the 1920’s
  - Text messages in Haitian Creole, Nepali, etc.

- Some have none:
  - Spanish-English travel guidebooks
  - Cookbooks

Examples of Domains

- Europarl
- – European parliamentary proceedings •
- News-commentary
- – Analysis of current affairs •
- Subtitles
- – Film subtitles
- TED
- – Short talks on technology, entertainment, design

Domain Adaptation

- Domain Adaptation:
  Build a system on one kind of data,
  and **adjust** it to apply to another.

  [Often: retune parameters on new task's data]

Domain Adaptation

- Domain Adaptation:
  Build a system on one kind of data,
  and **adjust** it to apply to another.

  [Often: retune parameters on new task's data]

- In Natural Language Processing (NLP)
  Train a system on some language data,
  retune & apply it to a specific-- different-- task.

  [e.g. Build a speech recognition system using recorded phone calls,
  then tune it to use as an airline reservation hotline.]

Domain Adaptation

- Statistical machine translation (SMT) training data doesn't always match desired tasks.
- Have bilingual:
    - Parliament proceedings
    - Newspaper articles
    - Web scrapings
- Want to translate:
    - Travel scenarios
    - Interesting talks
    - Facebook updates
- Sometimes want specificity, not just breadth!

Without Adaptation

- MT systems make errors in new domains [Carpuat et al, 2013]
- OOV words are a big problem

Without Adaptation

- MT systems make errors in new domains [Carpuat et al, 2013]
- OOV words are a big problem
- So are words with new senses
- Even known words with known translations can have wrong translation scores

Word Senses vs. Domains

- Many words have multiple senses
- Cross-lingual mapping difficult for *all* contexts
- Senses are often domain(?)-specific

Word Senses vs. Domains

- Fun with Tables and Chairs (into German):
  Table
  - Tisch -- General usage
  - Tabelle -- Tech usage
  Chair
  - Stuhl -- General usage
  - Vorsitzende -- Governmental usage

Word Senses vs. Domains: Contexts

- Table

The food is on the table = Das Essen ist auf dem Tisch.

The results are in the table = Die Ergebnisse sind in der Tabelle.

- Chair

He sat on the chair = Er saß auf dem Stuhl.

He is chair of the committee = Er ist Vorsitzender des Ausschusses.

The Easiest Thing

Given corpora A and B for different domains...

Concatenate A+B,
then train one system.

Advantage: No need to modify pipeline!

The Easiest Thing

Given corpora A and B for different domains...

Concatenate A+B,
then train one system.

Advantage: No need to modify pipeline!

NO.

The Easiest Thing (2)

Given corpora A and B for different domains...

Concatenate A+B, add a few more copies of B, then train one system.

The Easiest Thing (2)

Given corpora A and B for different domains...

Concatenate A+B, add a few more copies of B, then train one system.

STILL NO.

Phrase Table Augmentation

Often called "Fill-Up".
Adds *only* new entries [Nakov, 2008]

$$PT\_fillup = \{PT\_in\} \cup \{PT\_out - PT\_in\}$$

Phrase Table Augmentation

Often called "Fill-Up".
Adds *only* new entries [Nakov, 2008]

$$PT\_fillup = \{PT\_in\} \cup \{PT\_out - PT\_in\}$$

Better:
Add provenance feature as scaling factor
[Bisazza et al, 2011]

Phrase Table Augmentation

Often called "Fill-Up".
Adds *only* new entries [Nakov, 2008]

PT_fillup = {PT_in} ∪ {PT_out – PT_in}

Better:
Add provenance feature as scaling factor
[Bisazza et al, 2011]

Available in Moses: "combine-ptables"

Phrase Table Interpolation

Linear interpolation:
Works ok. Very easy with one weight per table.
[Finch and Sumita, 2008] [Yasuda et al, 2008]

Phrase Table Interpolation

Linear interpolation:
Works ok. Very easy with one weight per table.
[Finch and Sumita, 2008] [Yasuda et al, 2008]

Better:
Interpolation weights set via perplexity minimization [Sennrich, 2011].

Phrase Table Interpolation

Linear interpolation:
Works ok. Very easy with one weight per table.
[Finch and Sumita, 2008] [Yasuda et al, 2008]

Better:
Interpolation weights set via perplexity minimization [Sennrich, 2011].

Available in Moses: "tmcombine"

Multiple Phrase Tables

Let the system figure it out!

Train multiple models, add extra features for the other tables, then tune. [Birch et al, 2007]

Laziest approach; works well.

Multiple Phrase Tables

Let the system figure it out!

Train multiple models, add extra features for the other tables, then tune. [Birch et al, 2007]

Laziest approach; works well.

Bonus: LM combination for free!

Multiple Phrase Tables

Downsides:

Models can't reinforce each other.

Extra features to tune.

Multiple Phrase Tables

Downsides:

Models can't reinforce each other.

Extra features to tune.

Available in Moses and cdec.

Instance Weighting

- Not all training sentences are equally valuable.
- Weight their counts accordingly:
  A good sentence's ngrams receive a count of 1, a not-so-good one gets fractional counts.
- Done at either sentence level [Matsoukas et al, 2009] or n-gram [Foster et al, 2010].

Instance Weighting

- Not all training sentences are equally valuable.
- Weight their counts accordingly:
  A good sentence's ngrams receive a count of 1, a not-so-good one gets fractional counts.
- Done at either sentence level [Matsoukas et al, 2009] or n-gram [Foster et al, 2010].
- Available in Moses:
  "Alternate Weight Settings" (sentence-level).

Data Selection

- Don't change the pipeline, improve the input!
- Not all sentences are equally valuable...
- For a particular translation task:
  - Identify the most relevant training data.
  - Build a model on only this subset.
- Goal:
  - Better task-specific performance
  - Cheaper (computation, size, time)

Data Selection Algorithm

- Quantify the domain
- Compute similarity of sentences in pool to the in-domain corpus
- Sort pool sentences by score
- Select top n%

Data Selection Algorithm

- Quantify the domain
- Compute similarity of sentences in pool to the in-domain corpus
- Sort pool sentences by score
- Select top n%
- Feed n% into normal pipeline to build task-specific SMT system
- Combine with system trained on in-domain data (optional)
- Apply task-specific system to task.

Data Selection Algorithm

- Quantify the domain
- Compute similarity of sentences in pool to the in-domain corpus
- Sort pool sentences by score
- Select top n%
- Feed n% into normal pipeline to build task-specific SMT system
- Combine with system trained on in-domain data (optional) <-- multiple model scenario!
- Apply task-specific system to task

Perplexity-Based Filtering

- A language model LM_Q measures the likelihood of some text by its perplexity:

$$ppl_{LM_Q}(s) = 2^{-\frac{1}{N}\sum_{i=1}^{N}\log LM_Q(w_i|h_i)} = 2^{H_{LM_Q}(s)}$$

- Intuition: Average branching factor of LM

Perplexity-Based Filtering

- A language model LM_Q measures the likelihood of some text by its perplexity:

$$ppl_{LM_Q}(s) = 2^{-\frac{1}{N}\sum_{i=1}^{N}\log LM_Q(w_i|h_i)} = 2^{H_{LM_Q}(s)}$$

- Intuition: Average branching factor of LM
- Cross-entropy H (of a text w.r.t. an LM) is log(*ppl*).

- Perplexity used as data selection similarity score [Yasuda et al, 2008]

Cross-Entropy Difference

- Perplexity-based filtering:
  Score sentences in pool by perplexity with in-domain LM.
  Then rank, select, etc.
- However:

  By construction, the data pool does not match the target task [Moore and Lewis, 2010]

Cross-Entropy Difference

- Score and rank by cross-entropy difference:

$$\underset{s \in POOL}{\text{argmin}} \quad H_{LM_{IN}}(s) - H_{LM_{POOL}}(s)$$

- Moore-Lewis biases towards sentences that both:
  - Are *like* the target task
  - Are *unlike* the pool average.

Bilingual Cross-Entropy Difference

- Extend the Moore-Lewis similarity score for use with bilingual data [Axelrod et al, 2011]:

$$(H_{L1}(s_1, LM_{IN}) - H_{L1}(s_1, LM_{POOL}))$$
$$+(H_{L2}(s_2, LM_{IN}) - H_{L2}(s_2, LM_{POOL}))$$

Bilingual Cross-Entropy Difference

- Extend the Moore-Lewis similarity score for use with bilingual data [Axelrod et al, 2011]:

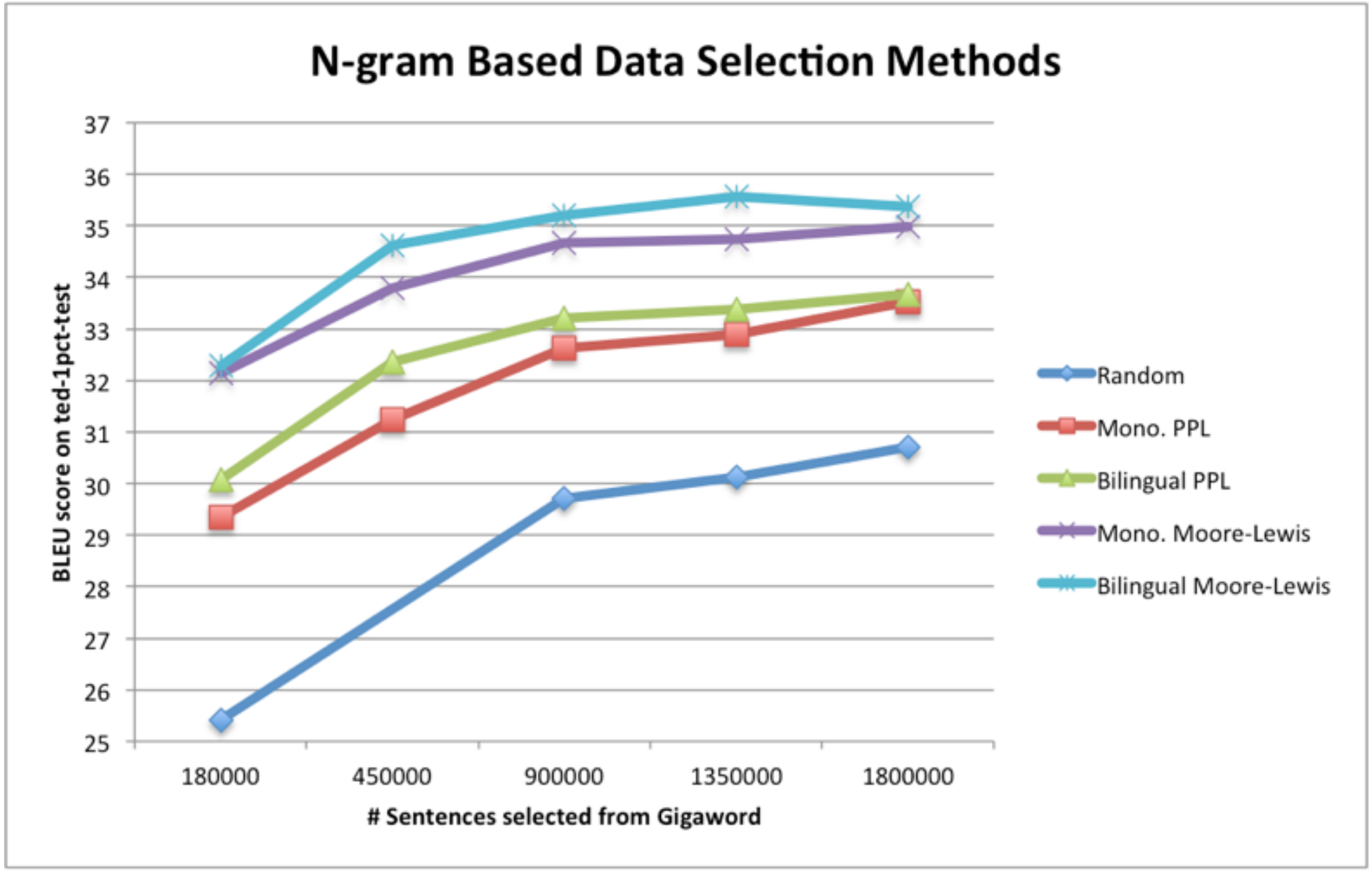
$$(H_{L1}(s_1, LM_{IN}) - H_{L1}(s_1, LM_{POOL}))$$
$$+(H_{L2}(s_2, LM_{IN}) - H_{L2}(s_2, LM_{POOL}))$$

- Downside:
  Find best n% via grid-search
  Not helpful for low-difference domains.

Bilingual Cross-Entropy Difference

- Available in Moses:
  "mml" ("Modified Moore-Lewis")

- Standalone tool: "XenC" from LIUM/Matecat.



- Cross-entropy difference more effective than perplexity-based and random baseline.
- Bilingual >> monolingual methods*

Bilingual Cross-Entropy Difference

- Bilingual >> monolingual methods*
  - * When there is a difference in morphological complexity between the languages.
  - * When there is enough data for a reasonable language model in both languages.

Similarity Ensembles

- Combine *de facto* standards:

  bilingual cross-entropy difference (SMT)

  tf-idf vector cosine similarity (IR)

  Levenshtein distance (string edit)
  [Wang et al, 2014]

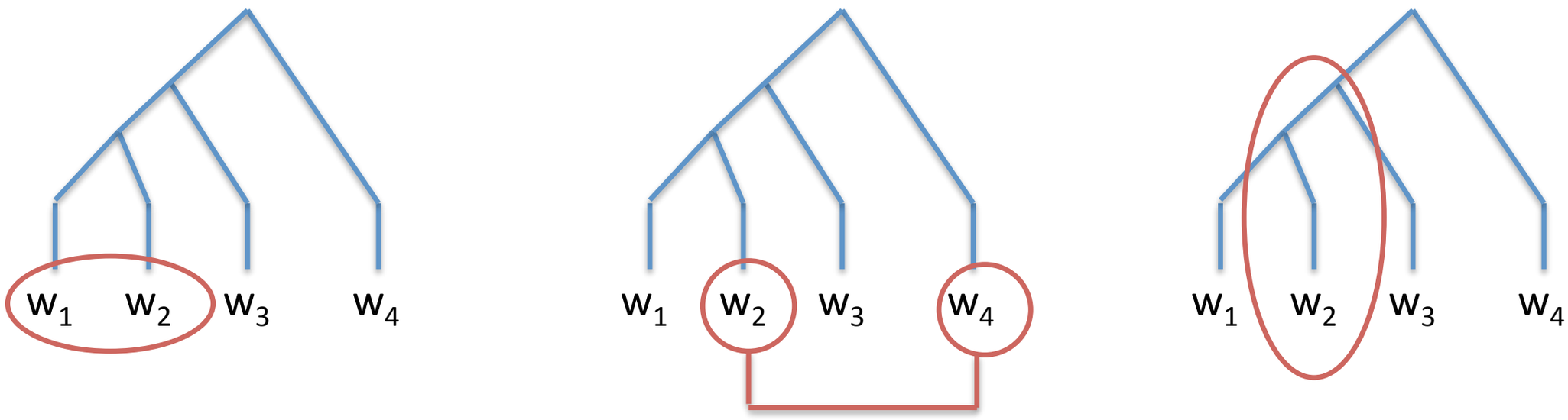
[This slide intentionally left blank]

Quantifying Textual Similarity

- Several facets of language along which we can quantify relevance:
  - domain
  - topic
  - style
- What the text in a translation task is for,
  What it's about,
  How it is expressed.
- These are not independent!

Difference Between Facets

- All use the surface words, but each captures a slightly different set of characteristics of text:



N-gram:
local dependencies

Topical:
Longer-range dependencies

Structural:
Deeper dependencies.

Topic-Based Motivation

- Using all the words in the corpora helped. ...but are all words equally important?
- Often assumed that content words of a corpus matter more.
  - A noun conveys more information than "the"
  - Thematic vocabulary is subset of overall vocabulary
  - Translating content words suffices to provide gist
- Topic models are one way to find them.

Topic Modeling

- A topic is a cluster of co-occurring words (not necessarily adjacently!) interpreted as a theme.
  - High-ranking words in TED topic model:
    - Technology: (design, computer, data, system)
    - Global: (Africa, dollars, business, market, food)
    - Planet: (water, earth, universe, ocean, trees)
    - Abstract: (life, love, god, stories, children)

Topic-Based Adaptation

- Incorporate additional topical features in decoder or phrase table. [Tam et al, 2007], [Gong, Zhou 2011], [Eidelman et al, 2012], [Hasler et al, 2014]

Close, But...

- Topic-based methods focused on thematic/ content/important words.
- Do not require task/pool-specific models.
- Good for tiny domains (50 sentences):
  Model used for similarity is trained on general data.

Close, But...

- Topic-based methods focused on thematic/ content/important words.
- Do not require task/pool-specific models.
- Good for tiny domains (50 sentences): Model used for similarity is trained on general data.
- **BUT** generally not better than n-gram based selection, which use all words.

New wrinkle

- Topic-based methods show importance of domain definition!
- Not only is not all data relevant, but the target domain is inherently not homogeneous.
- Potential impact of domain resolution.

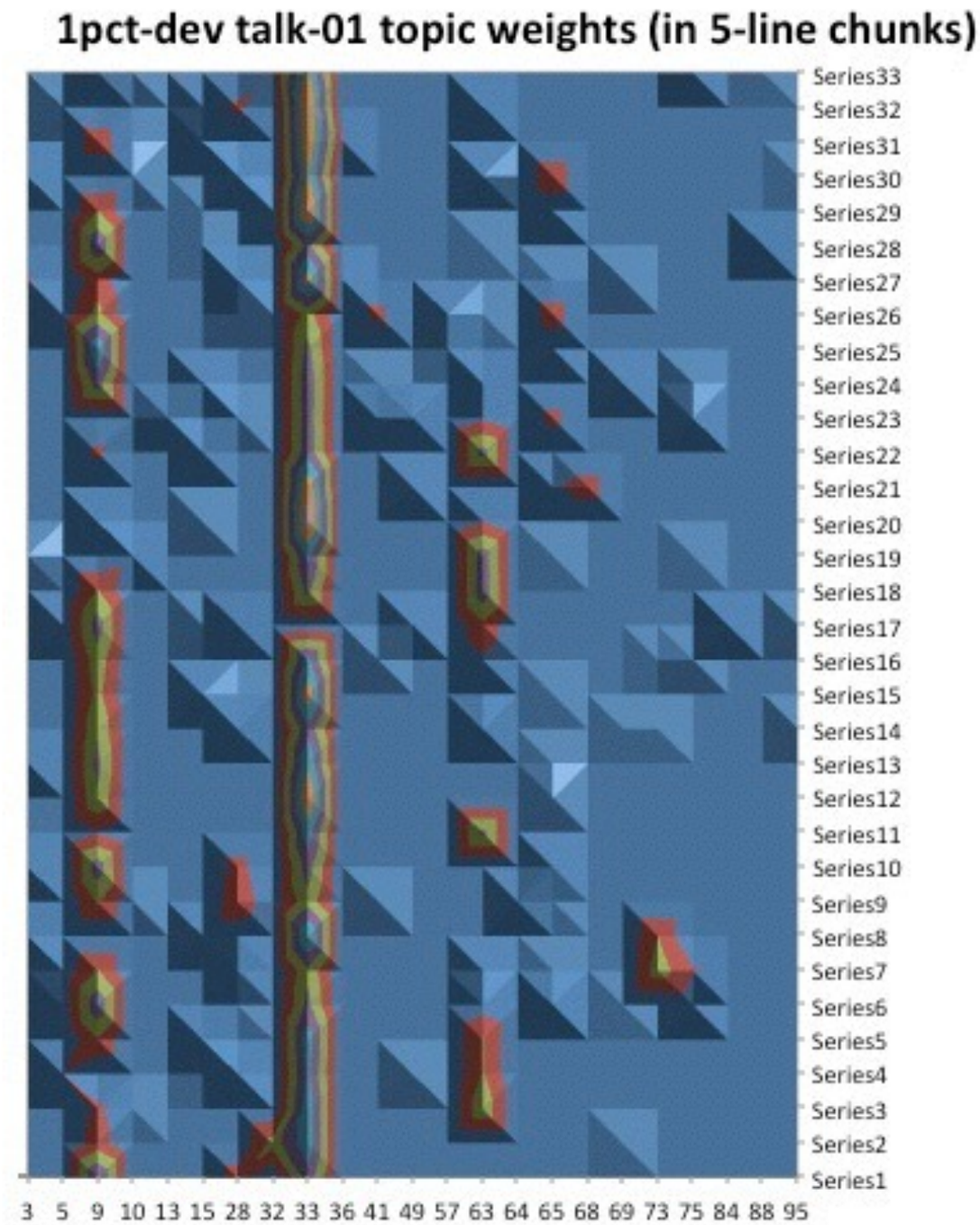
Topic Drift

Take a fine-grained view of target task!

Min 50 words per chunk; 5 lines suffices.

Topic distributions drift within a TED talk:

- Primary topic: #33
- Secondary topics: #9, #63



Secondary Topics

- Topic 9 keywords:
- space satellite systems earth csa canadian technology light radar radio mission station remote satellites science sun band field stars

- Topic 63 keywords:
- equipment vehicle vehicles building materials such material using designed air light design parts devices off systems construction hand side area motor

The Case of Topic 33

- Most significant topic for TED is topic #33.
- Keywords:
  ' s t my me like know re just don very us now because get here think going go ve want back then m say am way really much things good right said let lot something look today come even take why did ll little got great last down
- Few nouns and action verbs...
- Q: What is topic 33 about?

The Case of Topic 33

- Most significant topic for TED is topic #33.
- Keywords:
  ' s t my me like know re just don very us now because get here think going go ve want back then m say am way really much things good right said let lot something look today come even take why did ll little got great last down
- Few nouns and action verbs...
- Q: What is topic 33 about?
- A: Well...
- Topic 33 seems to signal what TED talks <u>are</u>: semi-formal, spoken, first-person presentations.

Style

- Describes social context of language
- Related to:
  register : variation between uses (not users)
  genre : communicative purpose
- Amount of attention paid to speech [Labov, 1984]
- Approximated by formality [Joos, 1961]
  1. Static: Wording never changes (Miranda warning).
  2. Formal: One-way, no interruptions (presentations).
  3. Consultative: Two-way (doctor/patient).
  4. Casual: In-group friends. Slang, interruptions.
  5. Intimate: Private, wording/grammar less important.

Style in NLP

- Stylometry / Authorship Attribution
- Classification problems:
  match authors, group authors by gender, etc.
- Stylistic features claimed independent of topic
- Mostly lexical, frequency-based. e.g. who uses "and" how often, and where. [Mosteller and Wallace, 1964]
- Syntactic features help [Argamon et al 1998]
- syntactic structure = proxy for style

Stylistic Similarity for MT

- Stylistic similarity should be useful when the medium differs between corpora:
- Twitter,
  Facebook status updates,
  SMS
  Speech transcripts
- How to use it is still open.

[This slide intentionally left blank]