Unsupervised Extraction of False Friends from Parallel Bi-Texts Using the Web as a Corpus

Preslav Nakov, Sofia University "St. Kliment Ohridski"
Svetlin Nakov, Sofia University "St. Kliment Ohridski"
Elena Paskaleva, Bulgarian Academy of Sciences

RANLP 2009 – September 12-18, 2009, Borovets, Bulgaria

Introduction

Introduction

- Cognates and False Friends
  - *Cognates* are pairs of words in different languages that are perceived as similar and are translations of each other
  - *False friends* are pairs of words in two languages that are perceived as similar, but differ in meaning
- The problem
  - Design an algorithm for extracting all pairs of false friends from a parallel bi-text

Cognates and False Friends

- Some cognates
  - *ден* in Bulgarian = *день* in Russian *(day)*
  - *idea* in English = *идея* in Bulgarian *(idea)*
- Some false friends
  - *майка* in Bulgarian *(mother)* ≠ *майка* in Russian *(vest)*
  - *prost* in German *(cheers)* ≠ *прост* in Bulgarian *(stupid)*
  - *gift* in German *(poison)* ≠ *gift* in English *(present)*

Method

Method

- False friends extraction from a parallel bi-text works in two steps:
  1. Find candidate cognates / false friends
     - Modified orthographic similarity measure
  2. Distinguish cognates from false friends
     - Sentence-level co-occurrences
     - Word alignment probabilities
     - Web-based semantic similarity
     - Combined approach

Step 1:
Identifying Candidate Cognates

Step 1: Finding Candidate Cognates

- Extract all word pairs (w_1, w_2) such that
  - $w_1 \in$ first language
  - $w_2 \in$ second language
- Calculate a *modified minimum edit distance ratio* MMEDR(w_1, w_2)
  - Apply a set of transformation rules and measure a weighted Levenshtein distance
- Candidates for cognates are pairs (w_1, w_2) such that
  - MMEDR$(w_1, w_2) > \alpha$

Step 1: Finding Candidate Cognates

Orthographic Similarity: MEDR

- **Minimum Edit Distance Ratio (MEDR)**
  - **MED(s_1, s_2) = the minimum number of INSERT / REPLACE / DELETE operations for transforming s_1 to s_2**
  - **MEDR**

$$\mathrm{MEDR}(s_1, s_2) = 1 - \frac{\mathrm{MED}(s_1, s_2)}{\max(|s_1|, |s_2|)}$$

  - **MEDR is also known as *normalized edit distance* (NED)**

Step 1: Finding Candidate Cognates

Orthographic Similarity: MMEDR

- Modified Minimum Edit Distance Ratio (MMEDR) for Bulgarian / Russian
  1. Transliterate from Russian to Bulgarian
  2. Lemmatize
  3. Replace some Bulgarian letter-sequences with Russian ones (e.g. strip some endings)
  4. Assign weights to the edit operations

Step 1: Finding Candidate Cognates

The MMEDR Algorithm

- Transliterate from Russian to Bulgarian
  - Strip the Russian letters "ь" and "ъ"
  - Replace "э" with "е", "ы" with "и", …
- Lemmatize
  - Replace inflected wordforms with their lemmata
  - Optional step: performed or skipped
- Replace some letter-sequences
  - Hand-crafted rules
  - Example: remove the definite article in Bulgarian words (e.g. "ът", "ят")

Step 1: Finding Candidate Cognates

The MMEDR Algorithm (2)

- **Assign weights to the edit operations:**
  - **0.5-0.9 for vowel to vowel substitutions, e.g. 0.5 for *е* → *о***
  - **0.5-0.9 for some consonant-consonant substitutions, e.g. *с* → *з***
  - **1.0 for all other edit operations**
- **MMEDR Example: the Bulgarian *първият* and the Russian *первый (first)***
  - **Previous steps produce *първи* and *перви*, thus MMED = 0.5 (weight 0.5 for ъ → *о*)**

Step 2:
Distinguishing between Cognates and False Friends

Method

- Our method for false friends extraction from parallel bi-text works in two steps:
  1. Find candidate cognates / false friends
     - Modified orthographic similarity measure
  2. Distinguish cognates from false friends
     - **Sentence-level co-occurrences**
     - Word alignment probabilities
     - Web-based semantic similarity
     - Combined approach

Sentence-Level Co-occurrences

- Idea: cognates are likely to co-occur in parallel sentences (unlike false friends)
- Previous work - Nakov & Pacovski (2006):
  - $\#(w_{bg})$ – the number of Bulgarian sentences containing the word w_{bg}
  - $\#(w_{ru})$ – the number of Russian sentences containing the word w_{ru}
  - $\#(w_{bg}, w_{ru})$ – the number of aligned sentences containing w_{bg} and w_{ru}

$$F_6(w_{bg}, w_{ru}) = \frac{\#(w_{bg}, w_{ru}) + 1}{\max\left(\frac{\#(w_{bg})+1}{\#(w_{ru})+1}, \frac{\#(w_{ru})+1}{\#(w_{bg})+1}\right)}$$

New Formulas for Sentence-Level Co-occurrences

- **New formulas for measuring similarity based on sentence-level co-occurrences**

$$E_1(w_{bg}, w_{ru}) = \frac{(\#(w_{bg}, w_{ru}) + 1)^2}{(\#(w_{bg}) + 1)(\#(w_{ru}) + 1)}$$

$$E_2(w_{bg}, w_{ru}) = \frac{(\#(w_{bg}, w_{ru}) + 1)^2}{P \times Q}$$

where

$$P = \#(w_{bg}) - \#(w_{bg}, w_{ru}) + 1$$

$$Q = \#(w_{ru}) - \#(w_{bg}, w_{ru}) + 1$$

Method

- Our method for false friends extraction from parallel bi-text works in two steps:
  1. Find candidate cognates / false friends
     - Modified orthographic similarity measure
  2. Distinguish cognates from false friends
     - Sentence-level co-occurrences
     - Word alignment probabilities
     - Web-based semantic similarity
     - Combined approach

Word Alignments

- Measure the semantic relatedness between words that co-occur in aligned sentences
  - Build directed word alignments for the aligned sentences in the bi-text
    - Using IBM Model 4
  - Average the translation probabilities $Pr(W_{bg}|W_{ru})$ and $Pr(W_{bg}|W_{ru})$:

$$lex(w_{bg}, w_{ru}) = \frac{\Pr(w_{bg}|w_{ru}) + \Pr(w_{ru}|w_{bg})}{2}$$

- Drawback: words that never co-occur in corresponding sentences have *lex* = 0

Method

- Our method for false friends extraction from parallel bi-text works in two steps:

1. Find candidate cognates / false friends
   - Modified orthographic similarity measure
2. Distinguish cognates from false friends
   - Sentence-level co-occurrences
   - Word alignment probabilities
   - **Web-based semantic similarity**
   - Combined approach

Web-based Semantic Similarity

- What is *local context*?
  - Few words before and after the target word

> Same day delivery of $\boxed{\text{fresh}}$ **flowers**, $\boxed{\text{roses}}$, and unique gift baskets from our online $\boxed{\text{boutique}}$. **Flower** $\boxed{\text{delivery}}$ online by local florists for $\boxed{\text{birthday}}$ **flowers**.

- The words in the local context of a given word are semantically related to it
- Need to exclude *stop words*: prepositions, pronouns, conjunctions, etc.
  - Stop words appear in all contexts
- Need for a sufficiently large corpus

Web-based Semantic Similarity (2)

- Web as a corpus
  - The Web can be used as a corpus to extract the local context for a given word
    - The Web is the largest available corpus
    - Contains large corpora in many languages
  - A query for a word in Google can return up to 1,000 text snippets
    - The target word is given along with its local context: few words before and after it
    - The target language can be specified

Web-based Semantic Similarity (3)

- **Web as a corpus**
  - **Example: Google query for "*flower*"**

Flowers, Plants, Gift Baskets - 1-800-**FLOWERS**.COM - Your Florist ... **Flowers**, balloons, plants, gift baskets, gourmet food, and teddy bears presented by 1-800-**FLOWERS**.COM, Your Florist of Choice for over 30 years.
Margarita **Flowers** - Delivers in Bulgaria for you! - gifts, **flowers**, roses ... Wide selection of BOUQUETS, FLORAL ARRANGEMENTS, CHRISTMAS ECORATIONS, PLANTS, CAKES and GIFTS appropriate for various occasions. CREDIT cards acceptable.
Flowers, plants, roses, & gifts. **Flowers** delivery with fewer ... **Flowers**, roses, plants and gift delivery. Order **flowers** from ProFlowers once, and you will never use **flowers** delivery from florists again.

Web-based Semantic Similarity (4)

- Measuring semantic similarity
  - Given two words, their local contexts are extracted from the Web
    - A set of words and their frequencies
  - Lemmatization is applied
  - Semantic similarity is measured using these local contexts
    - Vector-space model: build frequency vectors
    - Cosine: between these vectors

Web-based Semantic Similarity (5)

- Example of contextual word frequencies

word: flower

word	count
fresh	217
order	204
rose	183
delivery	165
gift	124
welcome	98
red	87
...	...

word: computer

word	count
Internet	291
PC	286
technology	252
order	185
new	174
Web	159
site	146
...	...

Web-based Semantic Similarity (6)

- Example of frequency vectors

v_1: flower

#	word	freq.
0	alias	3
1	alligator	2
2	amateur	0
3	apple	5
...	...	...
4999	zap	0
5000	zoo	6

v_2: computer

#	word	freq.
0	alias	7
1	alligator	0
2	amateur	8
3	apple	133
...	...	...
4999	zap	3
5000	zoo	0

- Similarity = cosine(v_1, v_2)

Web-based Semantic Similarity:
Cross-Lingual Semantic Similarity

- **Given**
  - **two words in different languages L_1 and L_2**
  - **a bilingual glossary G of known translation pairs $\{p \in L_1, q \in L_2\}$**
- **Measure cross-lingual similarity as follows**
  1. **Extract the local contexts of the target words from the Web: $C_1 \in L_1$ and $C_2 \in L_2$**
  2. **Translate the local context** $C_1 \xrightarrow{G} C_1^*$
  3. **Measure the similarity between C_1^* and C_2**
     - ***vector-space model***
     - ***cosine***

Method

- Our method for false friends extraction from parallel bi-text works in two steps:
  1. Find candidate cognates / false friends
     - Modified orthographic similarity measure
  2. Distinguish cognates from false friends
     - Sentence-level co-occurrences
     - Word alignment probabilities
     - Web-based semantic similarity
     - Combined approach

Combined Approach

- Sentence-level co-occurrences
  - Problems with infrequent words
- Word alignments
  - Work well only when the statistics for the target words are reliable
  - Problems with infrequent words
- Web-based semantic similarity
  - Quite reliable for unrelated words
  - Sometimes assigns very low scores to highly-related word pairs
  - Works well for infrequent words
- We combine all three approaches by adding up their similarity values

Experiments and Evaluation

Evaluation: Methodology

- We extract all pairs of cognates / false friends from a Bulgarian-Russian bi-text:
  - MMEDR(w_1, w_2) > 0.90
  - 612 pairs of words: 577 cognates and 35 false friends
- We order the pairs by their similarity score
  - according to 18 different algorithms
- We calculate *11-point interpolated average precision* on the ordered pairs

Resources

- Bi-text
  - The first seven chapters of the Russian novel "Lord of the World" + its Bulgarian translation
  - Sentence-level aligned with MARK ALISTeR (using the Gale-Church algorithm)
  - 759 parallel sentences
- Morphological dictionaries
  - Bulgarian: 1M wordforms (70,000 lemmata)
  - Russian: 1.5M wordforms (100,000 lemmata)

Resources (2)

- Bilingual glossary
  - Bulgarian / Russian glossary
  - 3,794 pairs of translation words
- Stop words
  - A list of 599 Bulgarian stop words
  - A list of 508 Russian stop words
- Web as a corpus
  - Google queries for 557 Bulgarian and 550 Russian words
  - Up to 1,000 text snippets for each word

Algorithms

- BASELINE – word pairs in alphabetical order
- COOC – the sentence-level co-occurrence algorithm with formula F6
- COOC+L – COOC with lemmatization
- COOC+E1 – COOC with the formula E1
- COOC+E1+L – COOC with the formula E1 and lemmatization
- COOC+E2 – COOC with the formula E2
- COOC+E2+L – COOC with the formula E2 and lemmatization
- WEB+L – Web-based semantic similarity with lemmatization
- WEB+COOC+L – average of WEB+L and COOC+L
- WEB+E1+L – average of WEB+L and E1+L
- WEB+E2+L – average of WEB+L and E2+L
- WEB+SMT+L – average of WEB+L and translation probability
- COOC+SMT+L – average of COOC+L and translation probability
- E1+SMT+L – average of E1+L and translation probability
- E2+SMT+L – average of E2+L and translation probability
- WEB+COOC+SMT+L – average of WEB+L, COOC+L and translation probability
- WEB+E1+SMT+L – average of WEB+L, E1+L, and translation probability
- WEB+E2+SMT+L – average of WEB+L, E2+L and translation probability

Results

Algorithm	11-pt Average Precision
BASELINE	4.17%
E2	38.60%
E1	39.50%
COOC	43.81%
COOC+L	53.20%
COOC+SMT+L	56.22%
WEB+COOC+L	61.28%
WEB+COOC+SMT+L	61.67%
WEB+L	63.68%
E1+L	63.98%
E1+SMT+L	65.36%
E2+L	66.82%
WEB+SMT+L	69.88%
E2+SMT+L	70.62%
WEB+E2+L	76.15%
WEB+E1+SMT+L	76.35%
WEB+E1+L	77.50%
WEB+E2+SMT+L	78.24%

Conclusion and Future Work

Conclusion

- We improved the accuracy of the best known algorithm by nearly 35%
- Lemmatization is a must for highly-inflectional languages like Bulgarian and Russian
- Combining multiple information sources works much better than any individual source

Future Work

- Take into account the part of speech
  - e.g. a verb and a noun cannot be cognates
- Improve the formulas for the sentence-level approaches
- Improved Web-based similarity measure
  - e.g. only use context words in certain syntactic relationships with the target word
- New resources
  - Wikipedia, EuroWordNet, etc.
  - Large parallel bi-texts as a source of semantic information

Thank you!

Questions?