Meerkat Mafia: Multilingual and Cross-Level Semantic Textual Similarity systems

Abhay Kashyap, Lushan Han, Jennifer Sleeman, Taneeya Satyapanich, Sunil Gandhi and Tim Finin
University of Maryland, Baltimore County

Roberto Yus, University of Zaragoza
Outline

• Introduction

• LSA word similarity

• Word similarity to text similarity

• Task 10: English and Spanish text similarity

• Task 3: Cross-level text similarity

• Conclusion and future work
Semantic Text Similarity 101

How similar are the two sentences semantically on a scale of 0-5?

The mouse ate some cheese
Cheddar was eaten by a rat

Pearson’s Correlation

It’s a 4!

Close enough!
Overview

- Participated in two 2014 SemEval tasks
  - Task 3: Cross-level semantic similarity sub-tasks
  - Task 10: STS English and Spanish sub-tasks
- Used may common components: LSA word similarity, algorithms applying it to text similarity, NLP tools & resources, Web APIs, machine learning
- Results were very good
  - Task 3: #1 in Sentence-Phrase, Phrase-Word, Word-Sense; #2 in Paragraph-Sentence
  - Task 10: #2 in both English and Spanish similarity
• Introduction

• LSA word similarity

• Word similarity to text similarity

• Task 10: English and Spanish text similarity

• Task 3: Cross-level text similarity

• Conclusion and future work
Word Similarity Model

• Foundation of out systems is a robust model for word similarity

• Semantic similarity across POS categories
  o Verb *marry* is similar to noun *wife*

• Handle synonyms and antonyms
  o *Love* and *hate* are similar; *love* and *like* even more so

• LSA word similarity + WordNet

LSA Similarity

Similar words usually occur in similar contexts!

- A large clean corpus produces reliable word co-occurrence stats
- Used corpus from Stanford WebBase project
  - Feb 2007 crawl, 100M pages from > 50K websites
  - Removed non-English text
  - Segmented into paragraphs and deduplicated
- Result: 3B word corpus - [http://ebiq.org/r/351](http://ebiq.org/r/351)
LSA Similarity

• Count co-occurrences of marry_{VB} & wife_{NN}
  o POS tagging + lemmatization (Stanford POS)
  o Only content words: noun, verb, adjective, adverb

• Two window sizes: three or nine words
  o ± 1: More precise context but works only for same POS
  o ± 4: Allows for similarity computation for different POS

• Vocabulary: 22K words and phrases plus over 2K verb phrases from WordNet

• Final dimensions is a 29k x 29k matrix
LSA similarity: SVD transform

• Replace frequency counts by logs
• Apply singular value decomposition to term-term matrix
• Retain 300 largest singular values
• Compute LSA similarity for two words as cosine similarity between their word vectors
### TOEFL Synonym Evaluation

- Our LSA models is better on some tasks than Google’s *word2vec*.
- TOEFL synonym task: pick best synonym from a list of four for 80 words.

<table>
<thead>
<tr>
<th></th>
<th>± 1 model</th>
<th>± 4 model</th>
<th>word2vec</th>
</tr>
</thead>
<tbody>
<tr>
<td>Correctly answered</td>
<td>73</td>
<td>76</td>
<td>67</td>
</tr>
<tr>
<td>OOV words</td>
<td>halfheartedly</td>
<td>halfheartedly</td>
<td>tranquility</td>
</tr>
<tr>
<td>Accuracy</td>
<td>92.4%</td>
<td>96.2%</td>
<td>84.8%</td>
</tr>
</tbody>
</table>
Leveraging WordNet

• LSA’s limitation – Polysemy!
  ○ Words with many senses have lower similarity scores

• Solution? Use WordNet knowledge to improve the scores

• Boost scores by exploiting synset, hypernym, derivative and other relations.
• Introduction

• LSA word similarity

• **Word similarity to text similarity**

• Task 10: English and Spanish text similarity

• Task 3: Cross-level text similarity

• Conclusion and future work
From word to text similarity

- Basic *align and penalize* approach
  1. Align words to maximize LSA word similarity
  2. Compute average word similarity for pairs
  3. Penalize unaligned terms

- Preprocessing: POS tag, lemmatization, REs to identify number and dates, stopword removal

- Word similarity wrapper for numbers, time expressions, pronouns and OOV words
A&P: align and penalize

• Aligning function for term $t \in S$

$$g(t) = \arg\max_{t' \in \mathcal{S}'} \text{sim}^{\uparrow'}(t, t'^\uparrow)$$

• The direction-dependent, many-to-many mapping is useful for aligning several words to a single term, e.g. ‘people who write’ to ‘authors’

• We get the term alignment scores

$$T = \sum_{t \in \mathcal{S}_1} \text{sim}'(t, g(t)) / 2 \cdot |\mathcal{S}_1| + \sum_{t \in \mathcal{S}_2} \text{sim}'(t, g(t)) / 2 \cdot |\mathcal{S}_2|$$
A&P: align and penalize

• Aligned pairs with low scores shouldn’t add to overall alignment score
  o John loves *dogs*
  o John loves *sandwiches*

• Antonyms often occur together, but their *semantic* similarity < their *distributional* similarity
  o John *loves* talking
  o John *hates* talking

• Add a *penalty* for pairs poorly aligned or known antonyms

\[
sim(\text{dogs, sandwiches}) = 0.03
\]

\[
sim(\text{loves, hates}) = 0.45
\]
A&P: align and penalize

• Pairs with similarity scores < 0.05 penalized by

\[ \sum_{\{t, g(t)\}} (\text{sim}^\uparrow (t, g(t)) + w^\downarrow f(t).w^\downarrow p(t)) / 2 \cdot |S| \]

Where \( w^\downarrow f(t) \) inversely weights log frequency of the term and \( w^\downarrow p(t) \) weights by its POS tag.

• We used a set of 50K WordNet antonym pairs to adjust their weight to a nominal 0.5

• After summing the penalties as \( P \) we compute the final STS score as

\[ \text{STS} = S - P \]
A&P align and penalize example

Cheddar cheese was eaten by a rat
The little mouse ate some cheese

align 1 => 2
align 2 => 1

Good alignment scores
Bad alignment penalty

STS score = T - P
• Introduction

• LSA word similarity

• Word similarity to text similarity

• Task 10: English and Spanish text similarity

• Task 3: Cross-level text similarity

• Conclusion and future work
Task 10: English runs

- **Pairing words (#2)** used unsupervised A&P
- **SuperSaiyan** and **Hulk** used a Support Vector Regression system with special features in addition to pairing words results
  - **Hulk (#6)** used a generic model trained on 3750 sentence pairs (1500 MSRvid, 1500 MSRpar, 750 Headlines)
  - **Super Saiyan (#5)** used domain-specific training for OnWWN (1361 pairs) and Images (1500 MSRvid pairs) and the generic model for others
Supervised Approach Details

• Words aligned by highest similarity (>0.1)

• Pairs weighted by Google word frequency

• OOV words (e.g., copasetic)
  
  o Retrieve its definition from Wordnik and use the A&P score as the word similarity

• Named Entities: try to link to DBpedia entities to see if they co-refer
Supervised Approach

Given a sentence pair $S_1$ and $S_2$ and their corresponding tokens $T_1$ and $T_2$

\[
\text{score} = \text{HarmonicMean}(\text{sim}(S_1, S_2), \text{sim}(S_2, S_1))
\]

\[
\text{sim}(S_1, S_2) = \frac{\sum_{t \in T_1} \text{avg}(w(t), w(s)) \times \arg \max_{s \in T_2} \text{LSA}(t, s)}{\sum_{t \in T_1} w(t)}
\]

\[
w(t) = \ln\left(\frac{\sum_{w' \in C} \text{freq}(w')}{\text{freq}(w)}\right)
\]

where $C$ is the set of words in the corpus

freq(w) is the frequency of a word w in the corpus
### English Results

<table>
<thead>
<tr>
<th>Dataset</th>
<th>Pairing Words</th>
<th>Hulk</th>
<th>SuperSaiyan</th>
</tr>
</thead>
<tbody>
<tr>
<td>Deft-forum (450 pairs)</td>
<td>0.4711 (7) *</td>
<td>0.4495 (15)</td>
<td>0.4918 (4)</td>
</tr>
<tr>
<td>Deft-news (300 pairs)</td>
<td>0.7628 (8)</td>
<td><strong>0.7850 (1)</strong></td>
<td>0.7712 (3)</td>
</tr>
<tr>
<td>Headlines (750 pairs)</td>
<td>0.7597 (8)</td>
<td>0.7571 (9)</td>
<td><strong>0.7666 (2)</strong></td>
</tr>
<tr>
<td>Images (750 pairs)</td>
<td>0.8013 (7)</td>
<td>0.7896 (10)</td>
<td>0.7676 (18)</td>
</tr>
<tr>
<td>OnWN (750 pairs)</td>
<td><strong>0.8745 (1)</strong></td>
<td>0.7872 (18)</td>
<td>0.8022 (12)</td>
</tr>
<tr>
<td>Tweet-news (750 pairs)</td>
<td>0.7793 (2)</td>
<td>0.7571 (7)</td>
<td>0.7651 (4)</td>
</tr>
<tr>
<td><strong>Weighted mean</strong></td>
<td><strong>0.7605 (2)</strong></td>
<td><strong>0.7349 (6)</strong></td>
<td><strong>0.7410 (5)</strong></td>
</tr>
</tbody>
</table>

- Pairing continues to do very well
- Hulk trained on data from all genres and SuperSaiyan on genre-specific data
- Supervised systems will do much better with more training data
Task 10 Spanish in a nutshell

“Un cojín es una funda de tela [...]”
“A cushion is a fabric cover [...]”

“Una almohada es un cojín [...]”
“A pillow is a rectangular pad [...]”

Pairing words & Hulk

3.7!
Improving the simple idea

- Direct Spanish to English translations good, yielding moderate results with our systems:
  - Paring, rank #13
  - Hulk, rank #5
- Handle translation anomalies to improve scores
  - *Las costas o costa de un mar ...*
    - → *Costs or the cost of a sea ... 😞*
    - → *Coasts or the coast of a sea ... 😊*
  - *Una almohada es un cojín en forma rectangular ...*
    - → *A pillow is a rectangular pad ... 😞*
    - → *A pillow is a rectangular cushion ... 😊*
Improving direct translation

Google’s suggested translation

- **Spanish**: Las costas o costa de un mar, lago o extenso río es la tierra a lo largo del borde de estos.
- **English**: Costs or the cost of a sea, lake or wide river is the land along the edge of these.

• Generate alternatives by considering possible translations for each word and combining them
  - las costas => costs, coasts, the costs, the coasts
  - o => or
  - costa => cost, coast, shore
  - de un => of the
  - mar => sea, ocean, briny
  - ...
Too many possible translations

- Costs or the cost of a sea, lake or wide river is the land along the edge of these.
- Coasts or the cost of a sea, lake or wide river is the land along the edge of these.
- The coasts or the cost of a sea, lake or wide river is the land along the edge of these.
- The shores or the cost of a sea, lake or wide river is the land along the edge of these.

• Control combinatorics by (i) limiting alternatives 20 and (ii) using word translations with Google score > 65
  - News and Wikipedia tests went from 480 & 324 sentence pairs to 5756 & 1776
Scoring candidate pairs

- $I_1$ and $I_2$ (a pair of Spanish sentences)
- Possible translations generated for each sentence:
  - $T_{I1} = \{T_{11}, T_{12}, T_{13}, \ldots, T_{1n}\}$
  - $T_{I2} = \{T_{21}, T_{22}, \ldots, T_{2m}\}$
- Compute similarity by using:

$$Sim_{SPA}(I_1, I_2) = \frac{\sum_{i=1}^{n} \sum_{j=1}^{m} Sim_{ENG}(T_{1i}, T_{2j})}{n \times m}$$
### Results on Spanish task

Three Spanish similarity runs submitted

<table>
<thead>
<tr>
<th>Dataset</th>
<th>Pairing Words</th>
<th>PairingAvg</th>
<th>Hulk</th>
</tr>
</thead>
<tbody>
<tr>
<td>Wikipedia (324 pairs)</td>
<td>0.6682 (12)</td>
<td>0.7431 (6)</td>
<td>0.7382 (8)</td>
</tr>
<tr>
<td>News (480 pairs)</td>
<td>0.7852 (12)</td>
<td><strong>0.8454 (1)</strong></td>
<td>0.8225 (6)</td>
</tr>
<tr>
<td><strong>Weighted mean</strong></td>
<td><strong>0.7380 (13)</strong></td>
<td><strong>0.8042 (2)</strong></td>
<td><strong>0.7885 (5)</strong></td>
</tr>
</tbody>
</table>

- Best run had weighted correlation of 0.8042, behind 1st place by only 0.003
- Wikipedia scores worse than news: more Spanish names with non-English characters that caused problems to our English-trained STS system
• Introduction
• LSA word similarity
• Word similarity to text similarity
• Task 10: English and Spanish text similarity
• Task 3: Cross-level text similarity
• Conclusion and future work
Task 3: Cross-level semantic similarity

• Did all four subtasks with good results
  o Paragraph-sentence: #2
  o Sentence-phrase: #1
  o Phrase-word: last 😞 / #1 * 😊
  o Word-sense: #1

• Reused LSA word similarity and (where appropriate) A&P

* Use of the wrong data file for phrase-word was only corrected after the deadline
• Treated like normal English sentence pairs (as in Task 10)
• Paragraph to Sentence – Rank 2 (Super Saiyan)
• Sentence to Phrase – Rank 1 (Super Saiyan)
Phrases to Word

- Ranked #1 for this task
- Word definitions from Wordnik
- Bing for documents with (i) word, (ii) phrase, and (iii) both
  Index in Lucene, calculate cos. similarity, produce features for
  similarity, mean & minimum
- Features used with a SVM regression model
Task 3: Word to sense subtask

• Ranked #1 for this task
• Used STS system to compare synonyms set of word both surface form and their definitions from WordNet with sense surface and sense’s definition
• Selected maximum score for each pair as feature
• Used additional Wordnik definitions for OOV words
• SVM Regression model trained and used to predict the final similarity score
Task 3: Word to sense subtask

- **WordNet**
  - Definition of sense

- **wordnik**
  - Definition of word
  - Definition of synset

- **STS system**
  - Maximum score of each pair

- **SVM Regression**
  - Similarity score
Cross-level STS Results

- Officially, none of our three runs did well overall
- Submission for word-phrase included the wrong data which was corrected after deadline. Original score of -0.044 was used in official run ranking.
- Without this error, SuperSaiyan would have been #1

<table>
<thead>
<tr>
<th>Dataset</th>
<th>Pairing Words</th>
<th>Hulk</th>
<th>Super Saiyan</th>
<th>Word Expand</th>
</tr>
</thead>
<tbody>
<tr>
<td>Paragraph-Sentence (500 pairs)</td>
<td>0.794 (10)</td>
<td>0.826 (4)</td>
<td>0.834 (2)</td>
<td></td>
</tr>
<tr>
<td>Sentence-Phrase (500 pairs)</td>
<td>0.704 (14)</td>
<td>0.705 (13)</td>
<td>0.777 (1)</td>
<td>0.457 (1)*</td>
</tr>
<tr>
<td>Phrase-Word (500 pairs)</td>
<td></td>
<td></td>
<td></td>
<td>0.389 (1)</td>
</tr>
<tr>
<td>Word-WordNet Sense (500 pairs)</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Weighted mean</td>
<td>0.7380 (13)</td>
<td>0.8042 (2)</td>
<td>0.7885 (5)</td>
<td></td>
</tr>
</tbody>
</table>
• Introduction

• LSA word similarity

• Word similarity to text similarity

• Task 10: English and Spanish text similarity

• Task 3: Cross-level text similarity

• Conclusion and future work
Conclusions

• The LSA word similarity model is the strength of our systems
• Our algorithms for computing text similarity are relatively simple and work well
• There are good sources of data to support specialized text similarity tasks, e.g. Wordnik
• Web search queries helped for OOV words
• Using Google translate worked surprisingly well for Spanish STS
Future Work

• Named entity recognition and matching
  E.g., "Obama" with "President Barack Obama”

• Syntactic features
  E.g., “Obama killed Osama” vs. “Osama killed Obama”

• Better and diverse training data for supervised approach

• Applications