Personalized PageRank over WordNet for Similarity and Word Sense Disambiguation

Eneko Agirre
e.agirre@ehu.es
(joint work with Aitor Soroa, some slides from Enrique Alfonseca)

University of the Basque Country
(Currently visiting Stanford)

Google, 2009
Introduction

Summary

- Present an integrated software based on Knowledge Bases (e.g. WordNet) for:
  - Similarity of word pairs
  - Disambiguate words with respect to knowledge base concepts (aka Word Sense Disambiguation)
- Excellent results (EACL, NAACL, IJCAI 2009)
- Open source: http://ixa2.si.ehu.es/ukb/
Outline

1. Introduction
2. WordNet, PageRank and Personalized PageRank
3. PPR for similarity [Agirre et al.2009b]
4. PPR for WSD [Agirre and Soroa2009]
5. PPR and WSD on specific domains [Agirre et al.2009a]
6. Conclusions
Measuring semantic similarity and relatedness are well studied problems in lexical semantics:

- Given two words or multiword-expressions, estimate how similar or related they are.
- Relatedness is a more general relationship, including topical relatedness or meronymy.
- Typically implemented as calculating a numeric value of similarity/relatedness.
## Similarity examples

<table>
<thead>
<tr>
<th>RG dataset</th>
<th>WordSim353 dataset</th>
</tr>
</thead>
<tbody>
<tr>
<td>cord smile</td>
<td>king cabbage</td>
</tr>
<tr>
<td>rooster voyage</td>
<td>professor cucumber</td>
</tr>
<tr>
<td>noon string</td>
<td>...</td>
</tr>
<tr>
<td>glass jewel</td>
<td>investigation effort</td>
</tr>
<tr>
<td>magician oracle</td>
<td>smart student</td>
</tr>
<tr>
<td>cushion pillow</td>
<td>...</td>
</tr>
<tr>
<td>cemetery graveyard</td>
<td>movie star</td>
</tr>
<tr>
<td>automobile car</td>
<td>journey voyage</td>
</tr>
<tr>
<td>midday noon</td>
<td>midday noon</td>
</tr>
<tr>
<td>gem jewel</td>
<td>fuck sex</td>
</tr>
<tr>
<td></td>
<td>tiger tiger</td>
</tr>
</tbody>
</table>
Two main approaches:

- Knowledge-based (Roget’s Thesaurus, WordNet, etc.)
- Corpus-based, also known as distributional similarity (co-occurrences)

Many potential applications, overcome brittleness (word match), specially in very short texts, information retrieval, textual entailment, machine translation.
Introduction

Similarity

- Two main approaches:
  - Knowledge-based (Roget’s Thesaurus, WordNet, etc.)
  - Corpus-based, also known as distributional similarity (co-occurrences)
- Many potential applications, overcome brittleness (word match), specially in very short texts, information retrieval, textual entailment, machine translation.


Goal: determine the senses of the words in a text.

- “... but the location on the south bank of the Thames estuary.”
- “... cash includes cheque payments, bank transfers ...”

Dictionary (e.g. WordNet):

- bank#1 sloping land, especially the slope beside a body of water.
- bank#2 a financial institution that accepts deposits and...
- bank#3 an arrangement of similar objects in row or in tiers.
- bank#4 a long ridge or pile.
- ... (10 senses total)

Many potential applications, enable natural language understanding, link text to knowledge base, deploy semantic web.
Goal: determine the senses of the words in a text.

- “... but the location on the south bank of the Thames estuary.”
- “... cash includes cheque payments, bank transfers ...”

Dictionary (e.g. WordNet):

- bank#1 sloping land, especially the slope beside a body of water.
- bank#2 a financial institution that accepts deposits and...
- bank#3 an arrangement of similar objects in row or in tiers.
- bank#4 a long ridge or pile.
- ... (10 senses total)

Many potential applications, enable natural language understanding, link text to knowledge base, deploy semantic web.
Goal: determine the senses of the words in a text.

- "... but the location on the south bank of the Thames estuary."
- "... cash includes cheque payments, bank transfers ..."

Dictionary (e.g. WordNet):

- bank#1 sloping land, especially the slope beside a body of water.
- bank#2 a financial institution that accepts deposits and...
- bank#3 an arrangement of similar objects in row or in tiers.
- bank#4 a long ridge or pile.
- ... (10 senses total)

Many potential applications, enable natural language understanding, link text to knowledge base, deploy semantic web.
Word Sense Disambiguation (WSD)

- Supervised corpus-based WSD performs best
  - Train classifiers on hand-tagged data (typically SemCor)
  - Data sparseness, e.g. *bank* 48 examples (25, 20, 2, 1, 0, …)
  - Results decrease when train/test from different sources (even Brown, BNC)
  - Decrease even more when train/test from different domains

- Knowledge-based WSD
  - Uses information in a KB (WordNet)
  - Performs close to but lower than Most Frequent Sense
  - Vocabulary coverage
  - Relation coverage
  - But …
Word Sense Disambiguation (WSD)

- Supervised corpus-based WSD performs best
  - Train classifiers on hand-tagged data (typically SemCor)
  - Data sparseness, e.g. *bank* 48 examples (25,20,2,1,0…)
  - Results decrease when train/test from different sources (even Brown, BNC)
  - Decrease even more when train/test from different domains

- Knowledge-based WSD
  - Uses information in a KB (WordNet)
  - Performs close to but lower than Most Frequent Sense
  - Vocabulary coverage
  - Relation coverage
  - But …
Domain adaptation

Deploying NLP techniques in real applications is challenging, specially for WSD:

- Sense distributions change across domains
- Data sparseness hurts more
- Context overlap is reduced
- New senses, new terms

But . . .

- Some words get less interpretations in domains: bank in finance, coach in sports
Introduction

Domain adaptation

Deploying NLP techniques in real applications is challenging, specially for WSD:

- Sense distributions change across domains
- Data sparseness hurts more
- Context overlap is reduced
- New senses, new terms

But... 

- Some words get less interpretations in domains: 
  *bank* in finance, *coach* in sports
Introduction

Similarity and WSD

If using knowledge-bases, both WSD and Similarity are closely intertwined:

- Similarity between words based on similarity between senses (implicitly doing disambiguation)
- WSD uses similarity of senses to context, or similarity between senses in context
Outline

1. Introduction
2. WordNet, PageRank and Personalized PageRank
3. PPR for similarity [Agirre et al.2009b]
4. PPR for WSD [Agirre and Soroa2009]
5. PPR and WSD on specific domains [Agirre et al.2009a]
6. Conclusions
Outline

1. Introduction
2. WordNet, PageRank and Personalized PageRank
3. PPR for similarity [Agirre et al.2009b]
4. PPR for WSD [Agirre and Soroa2009]
5. PPR and WSD on specific domains [Agirre et al.2009a]
6. Conclusions
Wordnet

- Most widely used hierarchically organized lexical database for English (Fellbaum, 1998)
- Broad coverage of nouns, verbs, adjectives, adverbs
- Main unit: synset (concept)
  - depository financial institution, bank#2, banking company
    a financial institution that accepts deposits and...
- Relations between concepts:
  - synonymy (built-in), hyperonymy, antonymy, meronymy, entailment, derivation, gloss
- Closely linked versions in several languages
Example of hypernym relations:

- bank
  - financial institution, financial organization
  - organization
    - social group
      - group, grouping
        - abstraction, abstract entity
          - entity

Representing WordNet as a graph:

- Nodes represent concepts
- Edges represent relations (undirected)
- In addition, directed edges from words to corresponding concepts (senses)
PageRank

- Given a graph, ranks nodes according to their relative structural importance.
- If an edge from $n_i$ to $n_j$ exists, a vote from $n_i$ to $n_j$ is produced.
  - Strength depends on the rank of $n_i$.
  - The more important $n_i$ is, the more strength its votes will have.
- PageRank can also be viewed as the result of a random walk process.
  - Rank of $n_i$ represents the probability of a random walk over the graph ending on $n_i$, at a sufficiently large time.
PageRank

- $G$: graph with $N$ nodes $n_1, \ldots, n_N$
- $d_i$: outdegree of node $i$
- $M$: $N \times N$ matrix

$$M_{ji} = \begin{cases} 
\frac{1}{d_i} & \text{an edge from } i \text{ to } j \text{ exists} \\
0 & \text{otherwise}
\end{cases}$$

PageRank equation:

$$Pr = cMPr + (1-c)v$$

- voting scheme
- a surfer randomly jumping to any node without following any paths on the graph

$c$: damping factor: the way in which these two terms are combined at each step
PageRank

- $G$: graph with $N$ nodes $n_1, \ldots, n_N$
- $d_i$: outdegree of node $i$
- $M$: $N \times N$ matrix

\[
M_{ji} = \begin{cases} 
\frac{1}{d_i} & \text{an edge from } i \text{ to } j \text{ exists} \\
0 & \text{otherwise}
\end{cases}
\]

PageRank equation:

\[
Pr = cMPr + (1 - c)v
\]

- voting scheme
- a surfer randomly jumping to any node without following any paths on the graph

$c$: damping factor: the way in which these two terms are combined at each step
PageRank

- $G$: graph with $N$ nodes $n_1, \ldots, n_N$
- $d_i$: outdegree of node $i$
- $M$: $N \times N$ matrix

$$M_{ji} = \begin{cases} 
\frac{1}{d_i} & \text{an edge from } i \text{ to } j \text{ exists} \\
0 & \text{otherwise}
\end{cases}$$

PageRank equation:

$$\text{Pr} \rightarrow cM\text{Pr} + (1 - c)v$$

- voting scheme
  - a surfer randomly jumping to any node without following any paths on the graph

$c$: damping factor: the way in which these two terms are combined at each step
PageRank

- $G$: graph with $N$ nodes $n_1, \ldots, n_N$
- $d_i$: outdegree of node $i$
- $M$: $N \times N$ matrix

$$M_{ji} = \begin{cases} 
\frac{1}{d_i} & \text{an edge from } i \text{ to } j \text{ exists} \\
0 & \text{otherwise}
\end{cases}$$

PageRank equation:

$$Pr \rightarrow cMPr + (1 - c)v$$

- voting scheme
- a surfer randomly jumping to any node without following any paths on the graph

$c$: damping factor: the way in which these two terms are combined at each step
PageRank

- $G$: graph with $N$ nodes $n_1, \ldots, n_N$
- $d_i$: outdegree of node $i$
- $M$: $N \times N$ matrix

$$M_{ji} = \begin{cases} 
\frac{1}{d_i} & \text{an edge from } i \text{ to } j \text{ exists} \\
0 & \text{otherwise}
\end{cases}$$

PageRank equation:

$$Pr \rightarrow cMPr + (1 - c)v$$

- voting scheme
- a surfer randomly jumping to any node without following any paths on the graph

$c$: damping factor: the way in which these two terms are combined at each step
Personalized PageRank

\[ \textbf{Pr} = cM\textbf{Pr} + (1 - c)\textbf{v} \]

- PageRank: \( \textbf{v} \) is a stochastic normalized vector, with elements \( \frac{1}{N} \)
  - Equal probabilities to all nodes in case of random jumps

- Personalized PageRank, non-uniform \( \textbf{v} \) [Haveliwala2002]
  - Assign stronger probabilities to certain kinds of nodes
  - Bias PageRank to prefer these nodes

- For ex. if we concentrate all mass on node \( i \)
  - All random jumps return to \( n_i \)
  - Rank of \( i \) will be high
  - High rank of \( i \) will make all the nodes in its vicinity also receive a high rank
  - Importance of node \( i \) given by the initial \( \textbf{v} \) spreads along the graph
**Personalized PageRank**

\[ \mathbf{Pr} = c \mathbf{MPr} + (1 - c) \mathbf{v} \]

- **PageRank**: \( \mathbf{v} \) is a stochastic normalized vector, with elements \( \frac{1}{N} \)
  - Equal probabilities to all nodes in case of random jumps
- **Personalized PageRank**, non-uniform \( \mathbf{v} \) [Haveliwala2002]
  - Assign stronger probabilities to certain kinds of nodes
  - Bias PageRank to prefer these nodes
- For ex. if we concentrate all mass on node \( i \)
  - All random jumps return to \( n_i \)
  - Rank of \( i \) will be high
  - High rank of \( i \) will make all the nodes in its vicinity also receive a high rank
  - Importance of node \( i \) given by the initial \( \mathbf{v} \) spreads along the graph
Outline

1. Introduction
2. WordNet, PageRank and Personalized PageRank
3. PPR for similarity [Agirre et al.2009b]
4. PPR for WSD [Agirre and Soroa2009]
5. PPR and WSD on specific domains [Agirre et al.2009a]
6. Conclusions
Based on [Hughes and Ramage2007]

- Given a pair of words \((w_1, w_2)\),
  - Initialize teleport probability mass on either \(w_1\) or \(w_2\).
  - Run PPR

- The similarity is given by the cosine of the two PPR vectors.

**Experiment settings:**

- Damping value \(c = 0.85\)
- Calculations finish after 30 iterations

**Variations for Knowledge Base:**

- MCR (WordNet 1.6, closely linked to Spanish WordNet) and WordNet 3.0
- All WordNet relations, All WN+gloss relations
Datasets

Rubenstein and Goodenough (1965)
- 80 word pairs, judged by 51 human subjects
- Scale 0 to 4 based on their similarity
- Redone for a subset by Miller and Charles (1991)

WordSim353 dataset:
- Finkelstein et al. (2002)
- 353 word pairs, each with 13-16 human judgments
- Annotators were asked to rate similarity and relatedness.

Results given by rank correlation of system output with human ratings (Spearman)
## Results

Competition with 1.6Twords distributional thesaurus in Google.

<table>
<thead>
<tr>
<th>Method</th>
<th>Window size</th>
<th>RG dataset</th>
<th>WordSim353 dataset</th>
</tr>
</thead>
<tbody>
<tr>
<td>MCR16</td>
<td>1</td>
<td>0.83</td>
<td>0.53 (0.56)</td>
</tr>
<tr>
<td>WN30</td>
<td>2</td>
<td>0.83</td>
<td>0.56 (0.58)</td>
</tr>
<tr>
<td>WN30g</td>
<td>3</td>
<td>0.83</td>
<td>0.66 (0.69)</td>
</tr>
<tr>
<td></td>
<td>4</td>
<td>0.85</td>
<td>0.60</td>
</tr>
<tr>
<td></td>
<td>5</td>
<td>0.80</td>
<td>0.58</td>
</tr>
<tr>
<td></td>
<td>6</td>
<td>0.75</td>
<td>0.58</td>
</tr>
<tr>
<td></td>
<td>7</td>
<td>0.72</td>
<td>0.57</td>
</tr>
<tr>
<td>CW</td>
<td>1</td>
<td>0.83</td>
<td>0.63</td>
</tr>
<tr>
<td></td>
<td>2</td>
<td>0.83</td>
<td>0.60</td>
</tr>
<tr>
<td></td>
<td>3</td>
<td>0.85</td>
<td>0.59</td>
</tr>
<tr>
<td></td>
<td>4</td>
<td>0.89</td>
<td>0.60</td>
</tr>
<tr>
<td></td>
<td>5</td>
<td>0.80</td>
<td>0.58</td>
</tr>
<tr>
<td></td>
<td>6</td>
<td>0.75</td>
<td>0.58</td>
</tr>
<tr>
<td></td>
<td>7</td>
<td>0.72</td>
<td>0.57</td>
</tr>
<tr>
<td>BoW</td>
<td>1</td>
<td>0.81</td>
<td>0.64</td>
</tr>
<tr>
<td></td>
<td>2</td>
<td>0.80</td>
<td>0.64</td>
</tr>
<tr>
<td></td>
<td>3</td>
<td>0.79</td>
<td>0.64</td>
</tr>
<tr>
<td></td>
<td>4</td>
<td>0.78</td>
<td>0.65</td>
</tr>
<tr>
<td>Syn</td>
<td>G1,D0</td>
<td>0.81</td>
<td>0.62</td>
</tr>
<tr>
<td></td>
<td>G2,D0</td>
<td>0.82</td>
<td>0.55</td>
</tr>
<tr>
<td></td>
<td>G3,D0</td>
<td>0.81</td>
<td>0.62</td>
</tr>
<tr>
<td>CW+</td>
<td>4; G1,D0</td>
<td>0.88</td>
<td>0.66</td>
</tr>
<tr>
<td>Syn</td>
<td>4; G2,D0</td>
<td>0.87</td>
<td>0.64</td>
</tr>
</tbody>
</table>
Results

Unknown words in WordNet

<table>
<thead>
<tr>
<th>Method</th>
<th>Spearman</th>
<th>Interval</th>
</tr>
</thead>
<tbody>
<tr>
<td>WN30</td>
<td>0.56 (0.58)</td>
<td>[0.48, 0.63]</td>
</tr>
<tr>
<td>WN30 ∪ th</td>
<td>0.58</td>
<td>[0.51, 0.65]</td>
</tr>
<tr>
<td>WN30g</td>
<td>0.66 (0.69)</td>
<td>[0.59, 0.71]</td>
</tr>
<tr>
<td>WN30g ∪ th</td>
<td>0.68</td>
<td>[0.62, 0.73]</td>
</tr>
</tbody>
</table>
# Results

State-of-the-art on MC (subset of RG)

<table>
<thead>
<tr>
<th>Method</th>
<th>Source</th>
<th>Spearman (MC)</th>
<th>Pearson (MC)</th>
</tr>
</thead>
<tbody>
<tr>
<td>(Sahami et al., 2006)</td>
<td>Web snippets</td>
<td>0.62 [0.32, 0.81]</td>
<td>0.58 [0.26, 0.78]</td>
</tr>
<tr>
<td>(Chen et al., 2006)</td>
<td>Web snippets</td>
<td>0.69 [0.42, 0.84]</td>
<td>0.69 [0.42, 0.85]</td>
</tr>
<tr>
<td>(Wu and Palmer, 1994)</td>
<td>WordNet</td>
<td>0.78 [0.59, 0.90]</td>
<td>0.78 [0.57, 0.89]</td>
</tr>
<tr>
<td>(Leacock et al., 1998)</td>
<td>WordNet</td>
<td>0.79 [0.59, 0.90]</td>
<td>0.82 [0.64, 0.91]</td>
</tr>
<tr>
<td>(Resnik, 1995)</td>
<td>WordNet</td>
<td>0.81 [0.62, 0.91]</td>
<td>0.80 [0.60, 0.90]</td>
</tr>
<tr>
<td>(Lin, 1998a)</td>
<td>WordNet</td>
<td>0.82 [0.65, 0.91]</td>
<td>0.83 [0.67, 0.92]</td>
</tr>
<tr>
<td>(Bollegala et al., 2007)</td>
<td>Web snippets</td>
<td>0.82 [0.64, 0.91]</td>
<td>0.83 [0.67, 0.92]</td>
</tr>
<tr>
<td>(Jiang and Conrath, 1997)</td>
<td>WordNet</td>
<td>0.83 [0.67, 0.92]</td>
<td>0.85 [0.69, 0.93]</td>
</tr>
<tr>
<td>(Jarmasz, 2003)</td>
<td>Roget’s</td>
<td>0.87 [0.73, 0.94]</td>
<td>0.87 [0.74, 0.94]</td>
</tr>
<tr>
<td>(Patwardhan et al., 2006)</td>
<td>WordNet</td>
<td>n/a</td>
<td>0.91</td>
</tr>
<tr>
<td>(Alvarez and Lim, 2007)</td>
<td>WordNet</td>
<td>n/a</td>
<td>0.91</td>
</tr>
<tr>
<td>(Yang and Powers, 2005)</td>
<td>WordNet</td>
<td>0.87 [0.73, 0.91]</td>
<td>0.92 [0.84, 0.96]</td>
</tr>
<tr>
<td>(Hughes et al., 2007)</td>
<td>WordNet</td>
<td>0.90</td>
<td>n/a</td>
</tr>
<tr>
<td>Personalized PageRank</td>
<td>WordNet</td>
<td>0.89 [0.77, 0.94]</td>
<td>n/a</td>
</tr>
<tr>
<td>Bag of words</td>
<td>Web corpus</td>
<td>0.85 [0.70, 0.93]</td>
<td>0.84 [0.69, 0.93]</td>
</tr>
<tr>
<td>Context window</td>
<td>Web corpus</td>
<td>0.88 [0.76, 0.95]</td>
<td>0.89 [0.77, 0.95]</td>
</tr>
<tr>
<td>Syntactic contexts</td>
<td>Web corpus</td>
<td>0.76 [0.54, 0.88]</td>
<td>0.74 [0.51, 0.87]</td>
</tr>
</tbody>
</table>
# Results

## State-of-the-art on WordSim 353

<table>
<thead>
<tr>
<th>Method</th>
<th>Source</th>
<th>Spearman</th>
</tr>
</thead>
<tbody>
<tr>
<td>[Strube and Ponzetto2006]</td>
<td>Wikipedia</td>
<td>0.19–0.48</td>
</tr>
<tr>
<td>[Jarmasz2003]</td>
<td>WordNet</td>
<td>0.33–0.35</td>
</tr>
<tr>
<td>[Jarmasz2003]</td>
<td>Roget’s</td>
<td>0.55</td>
</tr>
<tr>
<td>[Hughes and Ramage2007]</td>
<td>WordNet</td>
<td>0.55</td>
</tr>
<tr>
<td>[Finkelstein et al.2002]</td>
<td>Web corpus, WN</td>
<td>0.56</td>
</tr>
<tr>
<td>[Gabrilovich and Markovitch2007]</td>
<td>ODP</td>
<td>0.65</td>
</tr>
<tr>
<td>[Gabrilovich and Markovitch2007]</td>
<td>Wikipedia</td>
<td>0.75</td>
</tr>
<tr>
<td>Personalized PageRank</td>
<td>WordNet</td>
<td>0.66 (0.69)</td>
</tr>
</tbody>
</table>
Cross-lingual evaluation

Consider pairs of words from different languages. Can we predict the similarities?

- **WordNet-based method:**
  - English WordNet graph, crosslingual lexical entries in synsets.
  - Personalized PageRank is calculated in the same way

- **Contextual method:**
  - Get the top 5 translations of the non-English word into English using the Google Machine Translation system.
  - Generate the context vectors for those 5 translations separately.
  - Add the vectors.
  - The rest of the procedure is the same.

- **Evaluation:**
  - RG and WordSim353
  - One of the words in each pair translated into Spanish
Cross-lingual evaluation

<table>
<thead>
<tr>
<th>Dataset</th>
<th>Method</th>
<th>overall</th>
<th>Δ</th>
<th>interval</th>
</tr>
</thead>
<tbody>
<tr>
<td>RG</td>
<td>MCR16</td>
<td>0.78</td>
<td>-0.05</td>
<td>[0.66, 0.86]</td>
</tr>
<tr>
<td></td>
<td>WN30g</td>
<td>0.74</td>
<td>-0.09</td>
<td>[0.61, 0.84]</td>
</tr>
<tr>
<td></td>
<td>Bag of words</td>
<td>0.68</td>
<td>-0.23</td>
<td>[0.53, 0.79]</td>
</tr>
<tr>
<td></td>
<td>Context windows</td>
<td><strong>0.83</strong></td>
<td>-0.05</td>
<td>[0.73, 0.89]</td>
</tr>
<tr>
<td>WS353</td>
<td>MCR16</td>
<td>0.42 (0.53)</td>
<td>-0.11 (-0.03)</td>
<td>[0.34, 0.51]</td>
</tr>
<tr>
<td></td>
<td>WN30g</td>
<td><strong>0.58</strong> (0.67)</td>
<td>-0.07 (-0.02)</td>
<td>[0.51, 0.64]</td>
</tr>
<tr>
<td></td>
<td>Bag of words</td>
<td>0.53</td>
<td>-0.12</td>
<td>[0.45, 0.61]</td>
</tr>
<tr>
<td></td>
<td>Context windows</td>
<td>0.52</td>
<td>-0.11</td>
<td>[0.44, 0.59]</td>
</tr>
</tbody>
</table>
Outline

1. Introduction
2. WordNet, PageRank and Personalized PageRank
3. PPR for similarity [Agirre et al.2009b]
4. PPR for WSD [Agirre and Soroa2009]
5. PPR and WSD on specific domains [Agirre et al.2009a]
6. Conclusions
Knowledge-based WSD

- Use information in WordNet for disambiguation:
  - “... cash includes cheque payments, bank transfers ... ”

- Traditional approach [Patwardhan et al. 2007]:
  - Compare each target sense of bank with those of the words in the context
  - Using semantic relatedness between pairs of senses
  - Combinatorial explosion: each word disambiguated individually
    - \( \text{sim}(\text{bank}#1,\text{cheque}#1) + \text{sim}(\text{bank}#1,\text{cheque}#2) + \text{sim}(\text{bank}#1,\text{payment}#1) \ldots \)
    - \( \text{sim}(\text{bank}#2,\text{cheque}#1) + \text{sim}(\text{bank}#2,\text{cheque}#2) + \text{sim}(\text{bank}#2,\text{payment}#1) \ldots \)
    - \ldots

- Graph-based methods
  - Exploit the structural properties of the graph underlying WordNet
  - Find globally optimal solutions
  - Disambiguate large portions of text in one go
  - Principled solution to combinatorial explosion
Knowledge-based WSD

- Use information in WordNet for disambiguation:
  - “... cash includes cheque payments, bank transfers ...”

- Traditional approach [Patwardhan et al.2007]:
  - Compare each target sense of bank with those of the words in the context
  - Using semantic relatedness between pairs of senses
  - Combinatorial explosion: each word disambiguated individually

    \[ \text{sim}(\text{bank#1},\text{cheque#1}) + \text{sim}(\text{bank#1},\text{cheque#2}) + \text{sim}(\text{bank#1},\text{payment#1}) \ldots \]
    \[ \text{sim}(\text{bank#2},\text{cheque#1}) + \text{sim}(\text{bank#2},\text{cheque#2}) + \text{sim}(\text{bank#2},\text{payment#1}) \ldots \]
    \[ \ldots \]

- Graph-based methods
  - Exploit the structural properties of the graph underlying WordNet
  - Find globally optimal solutions
  - Disambiguate large portions of text in one go
  - Principled solution to combinatorial explosion
Using PageRank for WSD

- Given a graph representation of the LKB
- PageRank over the whole WordNet would get a context-independent ranking of word senses

We would like:
- Given an input text, disambiguate all open-class words in the input taking the rest as context

Two alternatives
1. Create a context-sensitive subgraph and apply PageRank over it [Navigli and Lapata 2007, Agirre and Soroa 2008]
2. Use **Personalized PageRank** over the complete graph, initializing $v$ with the context words
Using PageRank for WSD

- Given a graph representation of the LKB
- PageRank over the whole WordNet would get a context-independent ranking of word senses
- We would like:
  - Given an input text, disambiguate all open-class words in the input taking the rest as context
- Two alternatives
  1. Create a context-sensitive subgraph and apply PageRank over it [Navigli and Lapata2007, Agirre and Soroa2008]
  2. Use **Personalized PageRank** over the complete graph, initializing \( v \) with the context words
Using Personalized PageRank (Ppr and Ppr_w2w)

- For each word $W_i$, $i = 1 \ldots m$ in the context
  - Initialize $v$ with uniform probabilities over words $W_i$
    - Context words act as source nodes injecting mass into the concept graph
  - Run Personalized PageRank
  - Choose highest ranking sense for target word

- Problem of $Ppr$
  - Senses of the same word might be linked
  - Those senses would reinforce each other and receive higher ranks

- $Ppr_w2w$ alternative:
  - Let the surrounding words decide which concept associated to $W_i$ has more relevance
  - For each target word $W_i$, concentrate the initial probability mass in words surrounding $W_i$, but not in $W_i$ itself
  - Run Personalized PageRank for each word in turn (higher cost)
Using Personalized PageRank (Ppr and Ppr_w2w)

For each word $W_i$, $i = 1 \ldots m$ in the context
- Initialize $v$ with uniform probabilities over words $W_i$
  
  Context words act as source nodes injecting mass into the concept graph
- Run Personalized PageRank
- Choose highest ranking sense for target word

Problem of $Ppr$
- Senses of the same word might be linked
- Those senses would reinforce each other and receive higher ranks

$Ppr_w2w$ alternative:
- Let the surrounding words decide which concept associated to $W_i$ has more relevance
- For each target word $W_i$, concentrate the initial probability mass in words surrounding $W_i$, but not in $W_i$ itself
- Run Personalized PageRank for each word in turn (higher cost)
Using Personalized PageRank (Ppr and Ppr\_w2w)

For each word $W_i$, $i = 1 \ldots m$ in the context
- Initialize $v$ with uniform probabilities over words $W_i$
  - Context words act as source nodes injecting mass into the concept graph
- Run Personalized PageRank
- Choose highest ranking sense for target word

Problem of $Ppr$
- Senses of the same word might be linked
- Those senses would reinforce each other and receive higher ranks

$Ppr\_w2w$ alternative:
- Let the surrounding words decide which concept associated to $W_i$ has more relevance
- For each target word $W_i$, concentrate the initial probability mass in words surrounding $W_i$, but not in $W_i$ itself
- Run Personalized PageRank for each word in turn (higher cost)
Experiment setting

- Two datasets
  - Senseval 2 All Words (S2AW)
  - Senseval 3 All Words (S3AW)
- Both labelled with WordNet 1.7 tags
- Create input contexts of at least 20 words
  - Adding sentences immediately before and after if original too short
- PageRank settings:
  - Damping factor \((c)\): 0.85
  - End after 30 iterations
Results and comparison to related work (S2AW)

(Mihalcea, 2005) Pairwise Lesk between senses, then PageRank.

(Sinha & Mihalcea, 2007) Several similarity measures, voting, fine-tuning for each PoS. Development over S3AW.

(Tsatsaronis et al., 2007) Subgraph BFS over WordNet 1.7 and eXtended WN, then spreading activation.

* No statistical significance (small dataset).

<table>
<thead>
<tr>
<th>Senseval-2 All Words dataset</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>System</strong></td>
</tr>
<tr>
<td>Mih05</td>
</tr>
<tr>
<td>Sihna07</td>
</tr>
<tr>
<td>Tsatsa07</td>
</tr>
<tr>
<td>Ppr</td>
</tr>
<tr>
<td>Ppr_w2w</td>
</tr>
<tr>
<td>MFS</td>
</tr>
</tbody>
</table>
Results and comparison to related work (S2AW)

(Mihalcea, 2005) Pairwise Lesk between senses, then PageRank.

(Sinha & Mihalcea, 2007) Several similarity measures, voting, fine-tuning for each PoS. Development over S3AW.

(Tsatsaronis et al., 2007) Subgraph BFS over WordNet 1.7 and eXtended WN, then spreading activation.

* No statistical significance (small dataset).

<table>
<thead>
<tr>
<th>Senseval-2 All Words dataset</th>
</tr>
</thead>
<tbody>
<tr>
<td>System</td>
</tr>
<tr>
<td>--------</td>
</tr>
<tr>
<td>Mih05</td>
</tr>
<tr>
<td>Sihna07</td>
</tr>
<tr>
<td>Tsatsa07</td>
</tr>
<tr>
<td>Ppr</td>
</tr>
<tr>
<td>Ppr_w2w</td>
</tr>
<tr>
<td>MFS</td>
</tr>
</tbody>
</table>

Note: PPR for WSD [Agirre and Soroa2009]
Comparison to related work (S3AW)

(Mihalcea, 2005) Pairwise Lesk between senses, then PageRank.

(Sinha & Mihalcea, 2007) Several similarity measures, voting, fine-tuning for each PoS. Development over S3AW.

(Navigli & Lapata, 2007) Subgraph DFS(3) over WordNet 2.0 plus proprietary relations, several centrality algorithms.

(Navigli & Velardi, 2005) SSI algorithm on WordNet 2.0 plus proprietary relations. Uses MFS when undecided.

<table>
<thead>
<tr>
<th>System</th>
<th>All</th>
<th>N</th>
<th>V</th>
<th>Adj.</th>
<th>Adv.</th>
</tr>
</thead>
<tbody>
<tr>
<td>Mih05</td>
<td>52.2</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>Sihna07</td>
<td>52.4</td>
<td>60.5</td>
<td>40.6</td>
<td>54.1</td>
<td>100.0</td>
</tr>
<tr>
<td>Nav07</td>
<td>-</td>
<td>61.9</td>
<td>36.1</td>
<td>62.8</td>
<td>-</td>
</tr>
<tr>
<td>Ppr</td>
<td>56.1</td>
<td>62.6</td>
<td>46.0</td>
<td>60.8</td>
<td>92.9</td>
</tr>
<tr>
<td>Ppr_w2w</td>
<td>57.4</td>
<td>64.1</td>
<td>46.9</td>
<td>62.6</td>
<td>92.9</td>
</tr>
<tr>
<td>MFS</td>
<td>62.3</td>
<td>69.3</td>
<td>53.6</td>
<td>63.7</td>
<td>92.9</td>
</tr>
<tr>
<td>Nav05</td>
<td>60.4</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
</tr>
</tbody>
</table>
Comparison to related work (S3AW)

(Mihalcea, 2005) Pairwise Lesk between senses, then PageRank.
(Sinha & Mihalcea, 2007) Several similarity measures, voting, fine-tuning for each PoS. Development over S3AW.
(Navigli & Lapata, 2007) Subgraph DFS(3) over WordNet 2.0 plus proprietary relations, several centrality algorithms.
(Navigli & Velardi, 2005) SSI algorithm on WordNet 2.0 plus proprietary relations. Uses MFS when undecided.

<table>
<thead>
<tr>
<th>System</th>
<th>All</th>
<th>N</th>
<th>V</th>
<th>Adj.</th>
<th>Adv.</th>
</tr>
</thead>
<tbody>
<tr>
<td>Mih05</td>
<td>52.2</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>Sihna07</td>
<td>52.4</td>
<td>60.5</td>
<td>40.6</td>
<td>54.1</td>
<td>100.0</td>
</tr>
<tr>
<td>Nav07</td>
<td>-</td>
<td>61.9</td>
<td>36.1</td>
<td>62.8</td>
<td>-</td>
</tr>
<tr>
<td>Ppr</td>
<td>56.1</td>
<td>62.6</td>
<td>46.0</td>
<td>60.8</td>
<td>92.9</td>
</tr>
<tr>
<td>Ppr_w2w</td>
<td>57.4</td>
<td>64.1</td>
<td>46.9</td>
<td>62.6</td>
<td>92.9</td>
</tr>
<tr>
<td>MFS</td>
<td>62.3</td>
<td>69.3</td>
<td>53.6</td>
<td>63.7</td>
<td>92.9</td>
</tr>
<tr>
<td>Nav05</td>
<td>60.4</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
</tr>
</tbody>
</table>
Outline

1. Introduction

2. WordNet, PageRank and Personalized PageRank

3. PPR for similarity [Agirre et al.2009b]

4. PPR for WSD [Agirre and Soroa2009]

5. PPR and WSD on specific domains [Agirre et al.2009a]

6. Conclusions
Examples from **BNC**, **Sports** and **Finances** sections Reuters
- 41 nouns: salient in either domain or with senses linked to these domains
- Sense inventory: WordNet v. 1.7.1

300 examples for each of the **41 nouns**
- Roughly 100 examples from each word and corpus

Freely available
Methods

- What would happen if we apply PPR-based WSD to specific domains?

  - Personalized PageRank over context
    - 
    
  - Personalized PageRank over related words
    - Get related words from distributional thesaurus [Koeling et al.2005]
    - coach: manager, captain, player, team, striker, ...

- Experiments on BNC, Sports, Finance dataset:
  - Supervised: train MFS, SVM, k-NN on SemCor examples
  - Static PageRank
  - PPRank: Personalized PageRank (same damping factors, iterations)
    - Use context
    - 50 related words [Koeling et al.2005] (BNC, Sports, Finance)
Methods

- What would happen if we apply PPR-based WSD to specific domains?

- Personalized PageRank over **context**
  - “... has never won a league title as **coach** but took Parma to success...”

- Personalized PageRank over **related words**
  - Get related words from distributional thesaurus [Koeling et al.2005]
  - **coach**: manager, captain, player, team, striker, ...

- Experiments on BNC, Sports, Finance dataset:
  - Supervised: train MFS, SVM, \(k\)-NN on SemCor examples
  - Static PageRank
  - PPRank: Personalized PageRank (same damping factors, iterations)
    - Use context
    - 50 related words [Koeling et al.2005] (BNC, Sports, Finance)
Methods

What would happen if we apply PPR-based WSD to specific domains?

- Personalized PageRank over **context**
  - “... has never won a league title as coach but took Parma to success...”

- Personalized PageRank over **related words**
  - Get related words from distributional thesaurus [Koeling et al.2005]
  - **coach**: manager, captain, player, team, striker, ...

- Experiments on BNC, Sports, Finance dataset:
  - Supervised: train MFS, SVM, $k$-NN on SemCor examples
  - Static PageRank
  - PPRank: Personalized PageRank (same damping factors, iterations)
    - Use context
    - 50 related words [Koeling et al.2005] (BNC, Sports, Finance)
Methods

- What would happen if we apply PPR-based WSD to specific domains?

- Personalized PageRank over context
  - “… has never won a league title as coach but took Parma to success…”

- Personalized PageRank over related words
  - Get related words from distributional thesaurus [Koeling et al.2005]
  - coach: manager, captain, player, team, striker, …

- Experiments on BNC, Sports, Finance dataset:
  - Supervised: train MFS, SVM, k-NN on SemCor examples
  - Static PageRank
  - PPRank: Personalized PageRank (same damping factors, iterations)
    - Use context
    - 50 related words [Koeling et al.2005] (BNC, Sports, Finance)
Results

<table>
<thead>
<tr>
<th>Systems</th>
<th>BNC</th>
<th>Sports</th>
<th>Finances</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Baselines</strong></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Random</td>
<td>19.7</td>
<td>19.2</td>
<td>19.5</td>
</tr>
<tr>
<td>SemCor MFS</td>
<td>34.9</td>
<td>19.6</td>
<td>37.1</td>
</tr>
<tr>
<td>Static PRank</td>
<td>36.6</td>
<td>20.1</td>
<td>39.6</td>
</tr>
<tr>
<td><strong>Supervised</strong></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>SVM</td>
<td>38.7</td>
<td>25.3</td>
<td>38.7</td>
</tr>
<tr>
<td>$k$-NN</td>
<td>42.8</td>
<td>30.3</td>
<td>43.4</td>
</tr>
<tr>
<td><strong>Context</strong></td>
<td>PPRank</td>
<td>43.8</td>
<td>35.6</td>
</tr>
<tr>
<td>Related words</td>
<td>PPRank</td>
<td>37.7</td>
<td>51.5</td>
</tr>
<tr>
<td>[koeling et al. 2005]</td>
<td>40.7</td>
<td>43.3</td>
<td>49.7</td>
</tr>
<tr>
<td><strong>Skyline</strong></td>
<td>Test MFS</td>
<td>52.0</td>
<td>77.8</td>
</tr>
</tbody>
</table>

- Supervised (MFS, SVM, $k$-NN) very low (see test MFS)
- Static PageRank close to MFS
- PPRank on context: best for BNC (* for statistical significance)
- PPRank on related words: best for Sports and Finance and improves over Koeling et al., who use pairwise WordNet similarity.
Conclusions

Knowledge-based method for similarity and WSD
Based on Personalized PageRank
Exploits whole structure of underlying KB efficiently

Performance:
- Similarity: best WordNet, comparable with 1.6 Tword, slightly below ESA
- WSD: Best KB algorithm S2AW, S3AW, Domains datasets
- WSD and domains:
  - Better than supervised WSD for domains
  - Acquisition of terms and ontology enrichment feasible
  - Interest in fields like biomedicine, where ontologies exist
Conclusions

- Knowledge-based method for similarity and WSD
- Based on Personalized PageRank
- Exploits whole structure of underlying KB efficiently
- Performance:
  - Similarity: best WordNet, comparable with 1.6 Tword, slightly below ESA
  - WSD: Best KB algorithm S2AW, S3AW, Domains datasets
  - WSD and domains:
    - Better than supervised WSD for domains
    - Acquisition of terms and ontology enrichment feasible
    - Interest in fields like biomedicine, where ontologies exist
Conclusions

- Easily ported to other languages
  - Provides cross-lingual similarity
  - Only requirement of having a WordNet

- Publicly available at http://ixa2.si.ehu.es/ukb
  - Both programs and data
  - Including program to construct graphs from new KB (e.g. Wikipedia)
  - GPL license, open source, free
Conclusions

- Easily ported to other languages
  - Provides cross-lingual similarity
  - Only requirement of having a WordNet
- Publicly available at http://ixa2.si.ehu.es/ukb
  - Both programs and data
  - Including program to construct graphs from new KB (e.g. Wikipedia)
  - GPL license, open source, free
Conclusions

Personalized PageRank over WordNet for Similarity and Word Sense Disambiguation

Eneko Agirre
e.agirre@ehu.es
(joint work with Aitor Soroa, some slides from Enrique Alfonseca)

University of the Basque Country
(Currently visiting Stanford)

Google, 2009
E. Agirre and A. Soroa.  
2008.  
Using the multilingual central repository for graph-based word sense disambiguation.  
In *Proceedings of LREC ’08*, Marrakesh, Morocco.

E. Agirre and A. Soroa.  
2009.  
Personalizing pagerank for word sense disambiguation.  
In *Proceedings of EACL-09*, Athens, Greece.

E. Agirre, O. Lopez de Lacalle, and A. Soroa.  
2009a.  
Knowledge-Based WSD on Specific Domains: Performing better than Generic Supervised WSD.  
In *Proceedings of IJCAI*, Pasadena, USA.

E. Agirre, A. Soroa, E. Alfonseca, K. Hall, J. Kravalova, and M Pas.  
2009b.  
A study on similarity and relatedness using distributional and WordNet-based approaches.
In Proceedings of annual meeting of the North American Chapter of the Association of Computational Linguistics (NAAC), Boulder, USA, June.

2002.

E. Gabrilovich and S. Markovitch.
2007.

T. H. Haveliwala.
2002.

Thad Hughes and Daniel Ramage.
2007.
Lexical semantic relatedness with random graph walks.

M. Jarmasz.
2003.
Roget’s Thesaurus as a lexical resource for Natural Language Processing.

R. Koeling, D. McCarthy, and J. Carroll.
2005.
Domain-specific sense distributions and predominant sense acquisition.

R. Mihalcea.
2005.
Unsupervised large-vocabulary word sense disambiguation with graph-based algorithms for sequence data labeling.
Conclusions

In Proceedings of HLT05, Morristown, NJ, USA.

R. Navigli and M. Lapata.
2007.
Graph connectivity measures for unsupervised word sense disambiguation.
In IJCAI.

S. Patwardhan, S. Banerjee, and T. Pedersen.
2007.

R. Sinha and R. Mihalcea.
2007.
Unsupervised graph-based word sense disambiguation using measures of word semantic similarity.
In Proceedings of the IEEE International Conference on Semantic Computing (ICSC 2007), Irvine, CA, USA.

M. Strube and S.P. Ponzetto.
2006.