Reciclagem: Exploring Portuguese Lexical Knowledge-Bases in the ASSIN Task

Hugo Gonçalo Oliveira, Ana Oliveira Alves, Ricardo Rodrigues

{hroliv,ana,rmanuel}@dei.uc.pt

CISUC, Department of Informatics Engineering
University of Coimbra, Portugal

ASSIN@PROPOR 2016
Tomar, 13 July, 2016

Introduction

- **Unsupervised** approach to the ASSIN task

Introduction

- **Unsupervised** approach to the ASSIN task
- Rely **exclusively** on the exploitation of **external sources of lexical-semantic knowledge**
  - Heuristics based on known **semantic relations**

Introduction

- **Unsupervised** approach to the ASSIN task
- Rely **exclusively** on the exploitation of **external sources of lexical-semantic knowledge**
  - Heuristics based on known **semantic relations**
- Instead of selecting a single knowledge base, **test** several and their combination

Introduction

- **Unsupervised** approach to the ASSIN task
- Rely **exclusively** on the exploitation of **external sources of lexical-semantic knowledge**
  - Heuristics based on known **semantic relations**
- Instead of selecting a single knowledge base, **test** several and their combination
- Two main goals:
  1. Test whether an **unsupervised approach is enough** to compute semantic similarity
     - For English, knowledge-based approaches to other tasks rival with unsupervised approaches (e.g. WSD)

Introduction

- **Unsupervised** approach to the ASSIN task
- Rely **exclusively** on the exploitation of **external sources of lexical-semantic knowledge**
  - Heuristics based on known **semantic relations**
- Instead of selecting a single knowledge base, **test** several and their combination
- Two main goals:
  1. Test whether an **unsupervised approach is enough** to compute semantic similarity
     - For English, knowledge-based approaches to other tasks rival with unsupervised approaches (e.g. WSD)
  2. **Indirect comparison** of a set of **open** Portuguese lexical knowledge bases using ASSIN as a benchmark

Procedure

Given two sentences t and h...

1. **Pre-processing** (OpenNLP, LemPORT [Rodrigues et al., 2014]):
   - Tokenization
   - POS-tagging
   - Lematization

Procedure

Given two sentences *t* and *h*...

1. **Pre-processing** (OpenNLP, LemPORT [Rodrigues et al., 2014]):
    - Tokenization
    - POS-tagging
    - Lematization
2. Compute a **similarity** score between words in *t* and *h*
    - According to the knowledge base
    - Words are represented as a tuple *(token,POS,lemma)*

Exploited resources

PAPEL

- **PAPEL** [Gonçalo Oliveira et al., 2008], relations extracted from Porto Editora's *Dicionário da Língua Portuguesa*, using grammars based on regularities in the definitions;

Exploited resources

- **PAPEL** [Gonçalo Oliveira et al., 2008], relations extracted from Porto Editora's *Dicionário da Língua Portuguesa*, using grammars based on regularities in the definitions;
- **Dicionário Aberto** [Simões et al., 2012], relations extracted using the grammars of PAPEL;
- **Wikcionário.PT**, relations extracted using the grammars of PAPEL;

Exploited resources

- **PAPEL** [Gonçalo Oliveira et al., 2008], relations extracted from Porto Editora's *Dicionário da Língua Portuguesa*, using grammars based on regularities in the definitions;
- **Dicionário Aberto** [Simões et al., 2012], relations extracted using the grammars of PAPEL;
- **Wikcionário.PT**, relations extracted using the grammars of PAPEL;
- **TeP** [Maziero et al., 2008], thesaurus that groups words with their synonyms + antonymy relations;
- **OpenThesaurus.PT**, similar to the previous, but smaller and without antonymy;

Exploited resources

- **PAPEL** [Gonçalo Oliveira et al., 2008], relations extracted from Porto Editora's *Dicionário da Língua Portuguesa*, using grammars based on regularities in the definitions;
- **Dicionário Aberto** [Simões et al., 2012], relations extracted using the grammars of PAPEL;
- **Wikcionário.PT**, relations extracted using the grammars of PAPEL;
- **TeP** [Maziero et al., 2008], thesaurus that groups words with their synonyms + antonymy relations;
- **OpenThesaurus.PT**, similar to the previous, but smaller and without antonymy;
- **OpenWordNet-PT** [de Paiva et al., 2012], open Portuguese wordnet;
- **PULO** [Simões and Guinovart, 2014], another Portuguese wordnet, smaller than the previous.

Semantic network representation

- Knowledge bases used as **semantic networks** $N(W, C)$
  - $|W|$ words (nodes)
  - $|C|$ connections between words (edges)
    - Each with a semantic relation label (e.g. SINÓNIMO-DE, HIPERÓNIMO-DE, PARTE-DE, ...)
    - Triples *word*$_1$ related-to *word*$_2$ (e.g. *animal* HIPERÓNIMO-DE *cão*, *roda* PARTE-DE *carro*)

Semantic network representation

- Knowledge bases used as **semantic networks** $N(W, C)$
  - $|W|$ words (nodes)
  - $|C|$ connections between words (edges)
    - Each with a semantic relation label (e.g. SINÓNIMO-DE, HIPERÓNIMO-DE, PARTE-DE, ...)
    - Triples $word_1$ related-to $word_2$ (e.g. *animal* HIPERÓNIMO-DE *cão*, *roda* PARTE-DE *carro*)
- PAPEL is already in this format!

Semantic network representation

- Knowledge bases used as **semantic networks** $N(W, C)$
  - $|W|$ words (nodes)
  - $|C|$ connections between words (edges)
    - Each with a semantic relation label (e.g. SINÓNIMO-DE, HIPERÓNIMO-DE, PARTE-DE, ...)
    - Triples $word_1$ related-to $word_2$ (e.g. *animal* HIPERÓNIMO-DE *cão*, *roda* PARTE-DE *carro*)
- PAPEL is already in this format!
- Wordnets and synonymy thesauri were **converted**
  - Each pair of words in a synset resulted in a synonymy triple
  - A relation for each pair of words in two related synsets

Semantic network representation

- Knowledge bases used as **semantic networks** $N(W, C)$
  - $|W|$ words (nodes)
  - $|C|$ connections between words (edges)
    - Each with a semantic relation label (e.g. SINÓNIMO-DE, HIPERÓNIMO-DE, PARTE-DE, ...)
    - Triples *word*$_1$ related-to *word*$_2$ (e.g. *animal* HIPERÓNIMO-DE *cão*, *roda* PARTE-DE *carro*)
- PAPEL is already in this format!
- Wordnets and synonymy thesauri were **converted**
  - Each pair of words in a synset resulted in a synonymy triple
  - A relation for each pair of words in two related synsets
  - For instance...

{*porta, portão*} parte-de {*automóvel, carro, viatura*}

(*porta* SINÓNIMO-DE *portão*), (*automóvel* SINÓNIMO-DE *carro*),
(*automóvel* SINÓNIMO-DE *viatura*), (*carro* SINÓNIMO-DE *viatura*),
(*porta* PARTE-DE *automóvel*), (*porta* PARTE-DE *carro*), (*porta* PARTE-DE *viatura*),
(*portão* PARTE-DE *automóvel*), (*portão* PARTE-DE *carro*), (*portão* PARTE-DE *viatura*)

Combining semantic networks

- **CARTÃO** [Gonçalo Oliveira et al., 2011], relations extracted from three dictionaries: PAPEL + Dicionário Aberto + Wikcionário.PT

Combining semantic networks

- **CARTÃO** [Gonçalo Oliveira et al., 2011], relations extracted from three dictionaries: PAPEL + Dicionário Aberto + Wikcionário.PT
- **Todos**, all the triples from all the exploited resources
- **Redun2**, all the triples in at least two exploited resources

Combining semantic networks

- **CARTÃO** [Gonçalo Oliveira et al., 2011], relations extracted from three dictionaries: PAPEL + Dicionário Aberto + Wikcionário.PT
- **Todos**, all the triples from all the exploited resources
- **Redun2**, all the triples in at least two exploited resources
- **CONTO.PT** [Gonçalo Oliveira, 2016], fuzzy wordnet, w/ confidence degrees based on the redundancy in the exploited resources
  - Words have variable memberships to synsets
  - Synset connections also have a confidence degree

Similarity heuristics

Three different kinds of tested heuristics:

- Word **neighbourhoods** in the semantic networks

Similarity heuristics

Three different kinds of tested heuristics:

- Word **neighbourhoods** in the semantic networks
- Based on the **structure** of the semantic network

Similarity heuristics

Three different kinds of tested heuristics:

- Word **neighbourhoods** in the semantic networks
- Based on the **structure** of the semantic network
- Based on the membership to **fuzzy synsets**

Neighbourhood similarity

- Similarity between two sentences t and h
  - Each represented as a **set of words**, T and H.
  - T and H contain **all the words of each sentence** and their **adjacencies in the semantic network**.

$$
\begin{aligned}
Neigh(word) = & synonyms(word) \\
& \cup\, hypernyms(word) \\
& \cup\, hyponyms(word) \\
& \cup\, parts(word) \\
& \cup\, \ldots
\end{aligned}
$$

  - Neighbourhood can be restricted to a subset of relation types

Neighbourhood similarity

- Similarity between two sentences t and h
  - Each represented as a **set of words**, T and H.
  - T and H contain **all the words of each sentence** and their **adjacencies in the semantic network**.

$$
\begin{aligned}
Neigh(word) = & synonyms(word) \\
& \cup\, hypernyms(word) \\
& \cup\, hyponyms(word) \\
& \cup\, parts(word) \\
& \cup\, \ldots
\end{aligned}
$$

  - Neighbourhood can be restricted to a subset of relation types
- Similarity between T and H:

$$Sim_{max}(t,h) = \sum_{i=1}^{|t|} max\Big(Sim\big(Neighbours(T_i), Neighbours(H_j)\big)\Big) : H_j \in H$$

(alternatives were tested but this lead to the best results)

Neighbourhood similarity heuristics

Adaptations of the Lesk algorithm [Banerjee and Pedersen, 2003]:

$$Jaccard(A, B) = \frac{|Neigh(A) \cap Neigh(B)|}{|Neigh(A) \cup Neigh(B)|}$$

$$Overlap(A, B) = \frac{|Neigh(A) \cap Neigh(B)|}{min(|Neigh(A)|, |Neigh(B)|)}$$

$$Dice(A, B) = 2.\frac{|Neigh(A)| \cup |Neigh(B)|}{|Neigh(A)| + |Neigh(B)|}$$

Average distance

- Between each pair of words (p_t, p_h), such that $p_t \in t$ and $p_h \in h$
- $Similarity = \frac{1}{1+distance}$
  - Should have probably used the lowest distance...

Network structure heuristics

Average distance

- Between each pair of words (p_t, p_h), such that $p_t \in t$ and $p_h \in h$
- $Similarity = \frac{1}{1+distance}$
  - Should have probably used the lowest distance...

Personalized PageRank [Agirre and Soroa, 2009]

- Order the network nodes according to their structural relevance for each sentence:
  1. Each node is weighted: $\frac{1}{|F|}$, if it is a word in f, 0 otherwise;
  2. With the previous weights, PageRank is run for 30 iterations;
  3. Nodes are ordered according to their rank;
  4. Define sets E_{fn} with the top-n words ($n = 50$).
  5. Similarity given by $\frac{E_{tn} \cap E_{hn}}{n}$

Network structure heuristics

Average distance

- Between each pair of words (p_t, p_h), such that $p_t \in t$ and $p_h \in h$
- $Similarity = \frac{1}{1+distance}$
  - Should have probably used the lowest distance...

Personalized PageRank [Agirre and Soroa, 2009]

- Order the network nodes according to their structural relevance for each sentence:
  1. Each node is weighted: $\frac{1}{|F|}$, if it is a word in f, 0 otherwise;
  2. With the previous weights, PageRank is run for 30 iterations;
  3. Nodes are ordered according to their rank;
  4. Define sets E_{fn} with the top-n words ($n = 50$).
  5. Similarity given by $\frac{E_{tn} \cap E_{hn}}{n}$
- Much tuning required to set the best parameters...

Fuzzy wordnet heuristics

Different approach, given the features of CONTO.PT...

- $\mu(w, S)$: **membership** of words w to synset S
- $conf(S_1, R, S_2)$: **confidence** on relation of type R between S_1 and S_2
- **Weights** $\rho_s > \rho_h > \rho_o$ for synonymy, hypernymy and other relations

Fuzzy wordnet heuristics

Different approach, given the features of CONTO.PT...

- $\mu(w, S)$: **membership** of words w to synset S
- $conf(S_1, R, S_2)$: **confidence** on relation of type R between S_1 and S_2
- **Weights** $\rho_s > \rho_h > \rho_o$ for synonymy, hypernymy and other relations

- $Sim(t, h)$ = maximum similarity between each pair of words (p_t, p_h), such that $p_t \in t$ and $p_h \in h$
  1. If there is at least one *synset* $S_{12} : p_1 \in S_{12} \wedge p_2 \in S_{12} \rightarrow Sim(p_1, p_2) = (\mu(p_1, S_1) + \mu(p_2, S_2)) \times \rho_s$
  2. If there are two *synsets* $S_1, S_2 : p_1 \in S_1 \wedge p_2 \in S_2 \wedge (S_1 \text{ relatedTo } S_2)$ $\rightarrow Sim(p_1, p_2) = (\mu(p_1, S_1) + \mu(p_2, S_2)) \times conf(S_1, R, S_2) \times \rho_{h/o}$

Fuzzy wordnet heuristics

Different approach, given the features of CONTO.PT...

- $\mu(w, S)$: **membership** of words w to synset S
- $conf(S_1, R, S_2)$: **confidence** on relation of type R between S_1 and S_2
- **Weights** $\rho_s > \rho_h > \rho_o$ for synonymy, hypernymy and other relations

- $Sim(t, h)$ = maximum similarity between each pair of words (p_t, p_h), such that $p_t \in t$ and $p_h \in h$
  1. If there is at least one *synset* $S_{12} : p_1 \in S_{12} \wedge p_2 \in S_{12} \rightarrow Sim(p_1, p_2) = (\mu(p_1, S_1) + \mu(p_2, S_2)) \times \rho_s$
  2. If there are two *synsets* $S_1, S_2 : p_1 \in S_1 \wedge p_2 \in S_2 \wedge (S_1 \text{ relatedTo } S_2)$ $\rightarrow Sim(p_1, p_2) = (\mu(p_1, S_1) + \mu(p_2, S_2)) \times conf(S_1, R, S_2) \times \rho_{h/o}$

- Not explored enough...

Entailment heuristics

Exclusively based on the fuzzy wordnet CONTO.PT...

- Use CONTO.PT as a normal wordnet by setting **cut-points**
  - θ_s, for synset memberships μ
  - θ_h, for hypernymy relations confidence *conf*

Entailment heuristics

Exclusively based on the fuzzy wordnet CONTO.PT...

- Use CONTO.PT as a normal wordnet by setting **cut-points**
  - θ_s, for synset memberships μ
  - θ_h, for hypernymy relations confidence *conf*

- δ is a predefined threshold
- $\Delta = ||T| - |H||$
- $if(\Delta < \delta)$
  - every word in T has a synonym in H
    return PARAPHRASE
  - every word in T has a synonym, a hypernym or a hyponym in H
    return ENTAILMENT
  - return NONE
- return NONE

Best results for similarity

Training

	Network	Heuristic	Pearson	MSE
PT-PT	Redun2	Overlap	0.600	1.173
	Redun2	Dice	0.598	1.185
	OpenWN-PT	Jaccard	0.596	1.159
	Redun2	Jaccard	0.596	1.190
	PAPEL	Overlap	0.594	1.195
	TeP	Dice	0.592	1.330
	PULO	Jaccard	0.590	1.259
	OpenWN-PT	PPR	0.528	1.301
	CONTO.PT	N/A	0.587	1.189
PT-BR	Redun2	Overlap	0.546	1.065
	OpenWN-PT	Dice	0.546	1.077
	OpenWN-PT	Jaccard	0.545	1.081
	OpenWN-PT	Overlap	0.544	1.039
	Redun2	Jaccard	0.544	1.070
	Redun2	Overlap	0.544	1.052
	PAPEL	Overlap	0.543	1.027
	TeP	Dice	0.543	1.090
	PULO	Jaccard	0.541	1.037
	PAPEL	PPR	0.447	1.150
	CONTO.PT	N/A	0.535	1.078

Best results

Test

	Network	Heuristic	Pearson	MSE
PT-PT	Redun2	Overlap	0.536	1.105
	Redun2	Dice	0.536	1.130
	Redun2	Jaccard	0.535	1.149
	OpenWN-PT	Jaccard	0.533	1.141
	TeP	Dice	0.532	1.131
	TeP	Jaccard	0.532	1.151
	PAPEL	Dice	0.530	1.146
	PULO	Jaccard	0.527	1.313
	OpenWN-PT	PPR	0.513	1.177
	CONTO.PT	N/A	0.526	1.179
PT-BR	TeP	Overlap	0.593	1.256
	OpenWN-PT	Dice	0.589	1.312
	OpenWN-PT	Overlap	0.589	1.345
	TeP	Dice	0.588	1.311
	OpenWN-PT	Jaccard	0.588	1.329
	Redun2	Dice	0.588	1.356
	PULO	Dice	0.584	1.326
	PAPEL	Dice	0.584	1.335
	OpenWN-PT	PPR	0.464	1.225
	CONTO.PT	N/A	0.580	1.367

COGNITIVE & MEDIA SYSTEMS CISUC · U C ·

Comments on Similarity

- Substantially **different results for training and test**
  - Training: best results for PT-PT (0.6 vs 0.54)
  - Test: best results for PT-BR (0.59 vs 0.53)

Comments on Similarity

- Substantially **different results for training and test**
  - Training: best results for PT-PT (0.6 vs 0.54)
  - Test: best results for PT-BR (0.59 vs 0.53)
- **Subtle differences** in Pearson among best configurations
  - Typically less than 0.01!

Comments on Similarity

- Substantially **different results for training and test**
  - Training: best results for PT-PT (0.6 vs 0.54)
  - Test: best results for PT-BR (0.59 vs 0.53)
- **Subtle differences** in Pearson among best configurations
  - Typically less than 0.01!
- Similar sentences share several words... are the **heuristics are more relevant than the semantic network**?
  - Best results always obtained with the Dice coefficient
  - PageRank always below neighbourhood-based heuristics
  - Average distance performed poorly

Comments on Similarity

- Substantially **different results for training and test**
  - Training: best results for PT-PT (0.6 vs 0.54)
  - Test: best results for PT-BR (0.59 vs 0.53)
- **Subtle differences** in Pearson among best configurations
  - Typically less than 0.01!
- Similar sentences share several words... are the **heuristics are more relevant than the semantic network**?
  - Best results always obtained with the Dice coefficient
  - PageRank always below neighbourhood-based heuristics
  - Average distance performed poorly
- Additional observations:
  - Redun2 was the best network, except for PT-BR test
    - Benefits of **combining knowledge from different sources**!
  - OpenWN-PT always close to the best
  - TeP got the best results in PT-BR test
  - CONTO.PT just slightly below the semantic networks

Best results

Entailment

	θ_s	θ_h	δ	**Accuracy**	**Macro F1**
PT-PT (train)	0.1	0.01	0.5	73.83%	0.45
	0.1	0.1	0.4	71.67%	0.38
	0.25	0.2	0.5	73.83%	0.45
PT-BR (train)	0.1	00.1	0.3	77.47%	0.31
	0.1	00.1	0.5	76.70%	0.42
	0.2	0.2	0.1	77.70%	0.29
PT-PT (test)	0.1	00.1	0.5	73.10%	0.43
	0.15	0.1	0.4	72.10%	0.38
	0.05	0.01	0.3	70.80%	0.32
PT-BR (test)	0.2	0.2	0.1	77.65%	0.29
	0.15	0.1	0.3	79.05%	0.39
	0.1	0.01	0.3	78.30%	0.33

- Higher accuracy in PT-BR, higher Macro F1 in PT-PT
- Gold collection
  - PT-PT: 24% entailment and 7% paraphrase
  - PT-BR: 17% entailment and 5% paraphrase

Conclusions

- **Modest results** when compared to other participants

- **Modest results** when compared to other participants
- Too many **parameters** involved
    - semantic networks, their combination, used relation types, relation weights, normalisation, from word similarity to sentence similarity, similarity measures, maximum distance, PageRank set size, cut points in fuzzy wordnet ...

Conclusions

- **Modest results** when compared to other participants
- Too many **parameters** involved
  - semantic networks, their combination, used relation types, relation weights, normalisation, from word similarity to sentence similarity, similarity measures, maximum distance, PageRank set size, cut points in fuzzy wordnet ...
- Difficult to explore / compare all of them properly
  - Genetic algorithm?
  - Lines for future work!

Conclusions

- **Modest results** when compared to other participants
- Too many **parameters** involved
  - semantic networks, their combination, used relation types, relation weights, normalisation, from word similarity to sentence similarity, similarity measures, maximum distance, PageRank set size, cut points in fuzzy wordnet ...
- Difficult to explore / compare all of them properly
  - Genetic algorithm?
  - Lines for future work!
- Computed scores used as **features** to the supervised approach ASAPP

Thank you!

Thank you!

Questions?

References I

Agirre, E. and Soroa, A. (2009).
Personalizing PageRank for word sense disambiguation.
In *Proceedings of the 12th Conference of the European Chapter of the Association for Computational Linguistics*, EACL'09, pages 33–41, Stroudsburg, PA, USA. ACL Press.

Banerjee, S. and Pedersen, T. (2003).
Extended gloss overlaps as a measure of semantic relatedness.
In *Proceedings of the 18th International Joint Conference on Artificial Intelligence (IJCAI'03)*, pages 805–810, CA, USA.

de Paiva, V., Rademaker, A., and de Melo, G. (2012).
OpenWordNet-PT: An open Brazilian wordnet for reasoning.
In *Proceedings of 24th International Conference on Computational Linguistics*, COLING (Demo Paper).

Gonçalo Oliveira, H. (2016).
CONTO.PT: Groundwork for the Automatic Creation of a Fuzzy Portuguese Wordnet.
In *Proceedings of 12th International Conference on Computational Processing of the Portuguese Language (PROPOR 2016)*, page in press, Tomar, Portugal. Springer.

Gonçalo Oliveira, H., Antón Pérez, L., Costa, H., and Gomes, P. (2011).
Uma rede léxico-semântica de grandes dimensões para o português, extraída a partir de dicionários electrónicos.
Linguamática, 3(2):23–38.

Gonçalo Oliveira, H., Santos, D., Gomes, P., and Seco, N. (2008).
PAPEL: A dictionary-based lexical ontology for Portuguese.
In *Proceedings of Computational Processing of the Portuguese Language – 8th International Conference (PROPOR 2008)*, volume 5190 of *LNCS/LNAI*, pages 31–40, Aveiro, Portugal. Springer.

Maziero, E. G., Pardo, T. A. S., Felippo, A. D., and Dias-da-Silva, B. C. (2008).
A Base de Dados Lexical e a Interface Web do TeP 2.0 - Thesaurus Eletrônico para o Português do Brasil.
In *VI Workshop em Tecnologia da Informação e da Linguagem Humana (TIL)*, pages 390–392.

Rodrigues, R., Gonçalo-Oliveira, H., and Gomes, P. (2014).
LemPORT: a High-Accuracy Cross-Platform Lemmatizer for Portuguese.
In Pereira, M. J. V., Leal, J. P., and Simões, A., editors, *Proceedings of the 3rd Symposium on Languages, Applications and Technologies (SLATE '14)*, OpenAccess Series in Informatics, pages 267–274, Germany. Schloss Dagstuhl — Leibniz-Zentrum für Informatik, Dagstuhl Publishing.

Simões, A. and Guinovart, X. G. (2014).
Bootstrapping a Portuguese wordnet from Galician, Spanish and English wordnets.
In *Advances in Speech and Language Technologies for Iberian Languages*, volume 8854 of *LNCS*, pages 239–248.

Simões, A., Álvaro Iriarte Sanromán, and Almeida, J. J. (2012).
Dicionário-Aberto: A source of resources for the Portuguese language processing.
In *Proceedings of 10th International Conference on the Computational Processing of the Portuguese Language (PROPOR 2012)*, volume 7243 of *LNCS*, pages 121–127, Coimbra Portugal. Springer.