An Evolutionary Game Theoretic Approach to Word Sense Disambiguation

Rocco Tripodi, Marcello Pelillo, Rodolfo Delmonte

Ca' Foscari University

October 27, 2014

1. Word Sense Disambiguation
2. Word Sense Disambiguation Games
3. Results

Word Sense Disambiguation

WSD definition

WSD is a task to identify the intended sense of a word in a computational manner based on the context in which it appears [Navigli, 2009].

- It has been studied since the beginning of NLP [Weaver, 1955] and also today is a central topic of this discipline.
- It is a central topic in applications like Text Entailment, Machine Translation, Opinion Mining and Sentiment Analysis.
- All of these applications require the disambiguation of ambiguous words, as preliminary process; otherwise they remain on the surface of the word, compromising the coherence of the data to be analyzed.

Word ambiguity: an example

Word ambiguity

The ambiguity of an individual word or phrase that can be used (in different contexts) to express two or more different meanings

- [...] one of the stars in the star cluster Pleiades [...]
- [...] one of the stars in the last David Lynch film [...]

Word ambiguity: an example

Word ambiguity

The ambiguity of an individual word or phrase that can be used (in different contexts) to express two or more different meanings

- [...] one of the stars in the star cluster Pleiades [...]
- a celestial body
- [...] one of the stars in the last David Lynch film [...]
- an actor who plays a principal role

WSD: a formal definition

- We can view a text T as a sequence of words $(w_1, w_2, ..., w_n)$
- WSD is the task of assigning the appropriate $sense(s)$ to all or some of the words in T
- identifying a mapping A from words to senses: $A(i) \subseteq Senses_D(w_i)$
- where $Senses_D(w_i)$ is the set of senses encoded in a dictionary D for word w_i
- and $A(i)$ is that subset of the senses of w_i which are appropriate in the context T
- WSD can be viewed as a classification task

WSD approaches

We can broadly distinguish three main approaches to WSD:

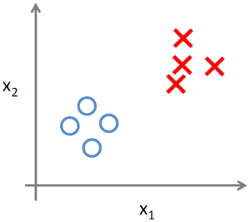
1. supervised methods
2. unsupervised methods
3. semi-supervised methods

Supervised approaches

An algorithm in which the classification model is built from examples which consists in:

1. an input feature space: X
2. an output label space: Y

The algorithm produce a mapping $f : X \to Y$ which should predict the correct output given a new input.



Supervised approaches: problems

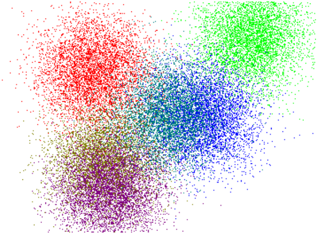
- The accuracy of supervised approaches is strongly dependent on the quantity of manually sense-tagged data available.
- The creation of such resources is extremely costly.
- As one would expect from Zipf's law, a substantial number of words will not occur in such resources.

Unsupervised approaches

An algorithm in which the classification model is built without examples, learning patterns in the input.

1. an input feature space: X
2. ~~an output label space: Y~~

The algorithm should find some intrinsic structures in the data.



Unsupervised approaches: graph based

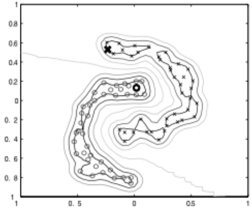
- Graph based methods use the notion of a co-occurrence graph: $G = (V, E)$
- where vertices V correspond to words in a text and edges E connect pairs of words which co-occur.
- By means of some similarity measure the edges of the graph are weighted $G = (V, E, w)$
- Then the vertices are clustered
- Each cluster represent a *semantic* domain which could be used for word sense induction or disambiguation

Semi-supervised approaches

An algorithm in which the classification model is built using large amount of unlabeled data, together with few labeled data, to build better classifiers.

1. an input feature space: X
2. an output label space: for few instances of X

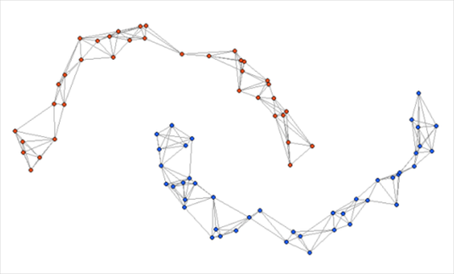
The algorithm requires less human effort and gives higher accuracy



Our approach: WSD games

Our approach to WSD is based on two fundamental principles:

1. the *homophily principle*
   Objects which are similar to each other are expected to have the same label [Easley and Kleinberg, 2010]
2. the *transductive learning*
   A semi-supervised learning technique which is used to propagate the class membership information from object to object



Game theory

- The outcomes of a person's decisions depend not just on how they choose among several options, but also on the choices made by the people with whom they interact.
- In order to maintain the **text coherence** we can see that the meaning of a word must by chosen according to the meaning of the other words in the text

Game definition

1. There is a set of participants, the **players**.
2. Each player has a set of options for how to behave (**strategies**)
3. For each choice of strategies, each player receives a **payoff** that can depend on the strategies selected by everyone

Dominant strategies – the prisoner's dilemma

When a player has a strategy that is strictly better than all other options, it is a strictly dominant strategy (DS).
We should expect that he or she will definitely play it.

p1/p2	**Not confess**	**Confess**
Not confess	-1 , -1	-10 , 0
Confess	0 , -10	-4 , -4

Confessing is a strictly DS. It is the best choice regardless of what the other player chooses.

Nash equilibrium

- If the players choose strategies that are best responses to each other, then no player has an incentive to deviate to an alternative strategy
- This concept is not one that can be derived purely from rationality on the part of the players; instead, it is an equilibrium concept.
- It is based on the believes of the players

p1/p2	**A**	**B**	**C**
A	4 , 4	0 , 2	0,2
B	0 , 0	1 , 1	0,2
C	0 , 0	0 , 2	1,1

Nodes/Players

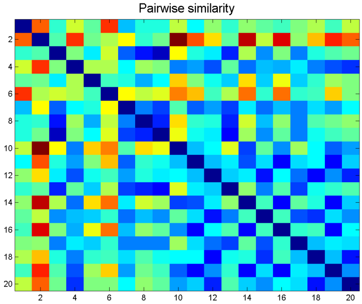
The players of the game are the target words x of our dataset X

$$X = \{x_i\}_{i=1}^{N} \tag{1}$$

where x_i corresponds to the i-th word to be disambiguated and N is the number of target words

Edges/Relatioins

From X we constructed the $N \times N$ similarity matrix W where each element w_{ij} is the similarity value assigned for the words i and j



Similarity measure

For our experiments we decided to use the following formula to compute the word similarities:

$$w_{ij} = Dice(x_i, x_j) \forall i, j \in X : i \neq j \tag{2}$$

where $Dice(x_i, x_j)$ is the Dice coefficient [Dice, 1945]. Which is computed as follows:

$$Dice(x_i, x_j) = \frac{2c(x_i, x_j)}{c(x_i) + c(x_j)} \tag{3}$$

where $c(x_i)$ is the total number of occurrences of x_i in a large corpus and $c(x_i, x_j)$ is the co-occurrence of the words x_i and x_j in the same corpus.

Player strategies/word senses

- For each player i, we use WordNet to collect its sense inventory $M_i = 1, \ldots, m$, where m is the number of synsets associated to word i.
- Then create the set of all possible senses, $C = 1, \ldots, c$.
- And initialize the strategy space of each player with the following formula:

$$s_{ij} = \begin{cases} |M|^{-1}, & \text{if sense } j \text{ is in } M_i. \\ 0, & \text{otherwise.} \end{cases} \tag{4}$$

Strategy space of the game

We can now define the strategy space S of the game in matrix form as:

$$\begin{matrix} S_{i1} & S_{i2} & \cdots & S_{ic} \\ \vdots & \vdots & \ldots & \vdots \\ S_{n1} & S_{n2} & \cdots & S_{nc} \end{matrix}$$

where each row corresponds to the strategy space of a player and each column corresponds to a class. Formally it is a c-dimensional space defined as:

$$\Delta_i = \{\sum_{h=1}^{m} s_{ih} = 1, \text{ and } s_{ih} \geq 0 \text{ for all } h\} \tag{5}$$

An example with two words

- Words: *area*, *country*
- Use WordNet to get the sense inventories $M_i = 1, \dots, m$
- Obtain the set of all possible senses, $C = 1, \dots, c$.
- The two words have 6 and 5 synsets, with a synset in common
- The strategy space S will have 10 dimension:

	s_1	s_2	s_3	s_4	s_5	s_6	s_7	s_8	s_9	s_{10}
S_{area}	6^{-1}	6^{-1}	6^{-1}	6^{-1}	6^{-1}	6^{-1}	0	0	0	0
$S_{country}$	0	5^{-1}	0	0	0	0	5^{-1}	5^{-1}	5^{-1}	5^{-1}

s_2: *(n) area, country: a particular geographical region of indefinite boundary* WordNet 3.0

Computing Nash equilibria

As in [Erdem and Pelillo, 2012] we used the dynamic interpretation of Nash equilibria in which the game is played repeatedly, until the system converges

$$S_{ih}(t+1) = S_{ih}(t)\frac{u_i(e_i^h)}{u_i(s(t))} \tag{6}$$

the utility function indicates the most profitable strategy for each player and it is computed as follows:

$$u_i(e_i^h) = \sum_{j \in D_u} (A_{ij}, s_j)_h + \sum_{k=1}^{c} \sum_{J \in D_{l|k}} A_{ij}(h,k) \tag{7}$$

$$u_i(s) = \sum_{j \in D_u} s_i^t w_{ij} s_j + \sum_{k=1}^{c} \sum_{J \in D_{l|k}} s_i^t (A_{ij})_k \tag{8}$$

Matlab implementation

MATLAB implementation

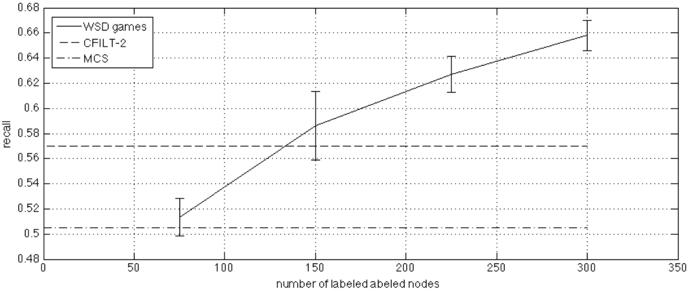
```
distance=inf;

while distance>epsilon

    old_x=x;

    x = x.*(A*x);

    x = x./sum(x);

    distance=pdist([x,old_x]');

end
```

Results

- SemEval-2 English all-words dataset [Agirre et al., 2009]
- Three documents
- 6000 word chunk
- $\approx$ 2000 target words
- The results have been provided by an analysis of their statistical significance 100 trials of randomly selected labeled points.

Results

The results are compared with Semeval10 best, CFILT-2 [Khapra et al., 2010] (recall 0.57), and with the most common sense (MCS) approach (recall 0.505).



Conclusion

- We have presented a new framework for WSD
- The framework is based on EGT
- It preserves the textual coherence
- It could be used for any language
- Preliminary experimental results demonstrate that our approach performs well compared with state-of-the-art algorithms.

Future work

We are implementing a new version of this algorithm in which:

- it will be used the semantic similarity among target words
- Not just the distributional similarity
- Include named entity disambiguation and linking

Bibliography

E. Agirre, O. L. De Lacalle, C. Fellbaum, A. Marchetti, A. Toral, and P. Vossen. Semeval-2010 task 17: All-words word sense disambiguation on a specific domain. In *Proceedings of the Workshop on Semantic Evaluations: Recent Achievements and Future Directions*, pages 123-128. Association for Computational Linguistics, 2009.

L. R. Dice. Measures of the amount of ecologic association between species. *Ecology*, 26(3): 297-302, 1945.

D. Easley and J. Kleinberg. Networks, crowds, and markets. *Cambridge University*, 2010.

A. Erdem and M. Pelillo. Graph transduction as a noncooperative game. *Neural Computation*, 24(3):700-723, 2012.

M. M. Khapra, S. Shah, P. Kedia, and P. Bhattacharyya. Domain-specific word sense disambiguation combining corpus based and wordnet based parameters. In *In 5th International Conference on Global Wordnet (GWC2010*. Citeseer, 2010.

R. Navigli. Word sense disambiguation: A survey. *ACM Computing Surveys (CSUR)*, 41 (2):10, 2009.

W. Weaver. Translation. *Machine translation of languages*, 14:15-23, 1955.