WSD-games: a Game-Theoretic Algorithm for Unsupervised Word Sense Disambiguation

Rocco Tripodi **Marcello Pelillo**
Ca' Foscari University of Venice
Via Torino 155
30172 Venezia, Italy
{rocco.tripodi, pelillo}@unive.it

Abstract

In this paper we present an unsupervised approach to word sense disambiguation based on evolutionary game theory. In our algorithm each word to be disambiguated is represented as a node on a graph and each sense as a class. The algorithm performs a consistent class assignment of senses according to the similarity information of each word with the others, so that similar words are constrained to similar classes. The dynamics of the system are formulated in terms of a non-cooperative multiplayer game, where the players are the data points to decide their class memberships and equilibria correspond to consistent labeling of the data.

1 Introduction

Word sense disambiguation (WSD) is the 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 it is a central topic of this discipline. Many algorithms have been proposed during the years, based on supervised (Zhong and Ng, 2010; Tratz et al., 2007), semi-supervised (Pham et al., 2005) and unsupervised (Mihalcea, 2005; McCarthy et al., 2007) learning models. Nowadays, even if supervised methods perform better in general domains, unsupervised and semi-supervised models are gaining attention from the research community with performances close to the state of the art (Ponzetto and Navigli, 2010). In particular Knowledge-based and graph based algorithms are emerging as interesting ways to face the problem (Agirre et al., 2009; Sinha and Mihalcea, 2007). The peculiarities of those algorithms are that they do not require any corpus evidence and use only the structural properties of a lexical database to perform the disambiguation task.

An unsupervised algorithm which has been implemented in different ways by the community (Mihalcea et al., 2004; Haveliwala, 2002; Agirre et al., 2014; De Cao et al., 2010) is the PageRank (Page et al., 1999). This algorithm is similar in spirit to ours but we instead of using the graph to compute the most important nodes (senses) in it, we use the network to model the geometry of the data and the interactions among the data points. In our system the nodes of the graph are interpreted as players, in the game theoretic sense (see Section 2), which play a game in order to maximize their utility. The concept of utility has been used in different ways in the game theory (GT) literature and in general it refers to the satisfaction that a player derives from the outcome of a game (Szabó and Fath, 2007). From our point of view increasing the utility of a word means increasing the textual coherence, in a distributional semantics perspective (Firth, 1957). In fact, in our framework a word always tries to chose a sense close to the senses which the other words in the text are likely to choose.

The starting point of our research is based on the assumption that the meaning of a sentence emerges from the interaction of the components which are involved in it. In our study we tried to model this interaction and to develop a system in which it is possible to map lexical items onto concepts. For this reason

Proceedings of the 9th International Workshop on Semantic Evaluation (SemEval 2015), pages 329–334,
Denver, Colorado, June 4-5, 2015.

we decided to use a powerful tool, derived from Evolutionary Game Theory (EGT): the non-cooperative games (see Section 2). EGT and GT have been used in different ways to study the language use (Pietarinen, 2007; Skyrms, 2010) and evolution (Nowak et al., 2001) but as far as we know, our is the first attempt to use it in a specific NLP task. This choice is motivated by the fact that GT models are able to perform a consistent labeling of the data (Hummel and Zucker, 1983; Pelillo, 1997), taking into account the contextual information. These features are of great importance for an unsupervised algorithm which tries to perform a WSD task, because them can be obtained without any supervision and help the system to adapt to different contextual domains.

2 Game Theory

In this section we briefly introduce some concepts of GT and EGT, for detailed analysis of these topics we refer to (Weibull, 1997; Leyton-Brown and Shoham, 2008; Sandholm, 2010).

GT provides predictive power in interactive decision situations. It has been introduced by Von Neumann and Morgenstern (1944) and in its normal form representation (which is the one we will use in our algorithm) it consists in: a finite set of players $I = (1, .., n)$, a set of pure strategies for each player $S_i = (s_1, ..., s_n)$ and an utility function $u_i : S_1 \times ... \times S_n \rightarrow \mathbb{R}$ which associates strategies to payoffs. The utility function depends on the combination of two strategies played together, not just on the strategy of a single player. An important assumption in GT is that the players are rational and try to maximize the value of u_i; furthermore in *non-cooperative games* the players choose their strategies independently. A strategy s_i^* is said to be *dominant* if and only if $u_i(s_i^*, s_{-i}) > u_i(s_i, s_{-i}), \forall s_{-i} \in S_{-i}$. As an example we can consider the famous *Prisoner's Dilemma* (in Table 1) where the strategy $confess$ is a *dominant strategy* for both players and this strategy combination is the *Nash equilibrium* of the game. Nash equilibria are those strategy profiles which are best response to the strategy of the co-player and no player has the incentive to unilaterally deviate from his strategy, because there is no way to do better.

1 \ 2	confess	don't confess
confess	-5,-5	0,-6
don't confess	-6,0	-1,-1

Table 1: The Prisoner's Dilemma.

2.1 Evolutionary Game Theory

EGT has been introduce by Smith and Price (1973) overcoming some limitations of traditional GT such as the hyper-rationality imposed on the players, in fact in real life situations the players choose a strategy according to heuristics or social norms (Szabó and Fath, 2007). Another important aspect of EGT is the introduction of an *inductive learning* process, in which the agents play the game repeatedly with their neighborhood, updating their believes on the state of the game and choosing their strategy accordingly. The strategy space of each player is defined as a probability distribution over its pure strategies. It is represented as a vector $x_i = (x_{i1}, \ldots, x_{im})$ where m is the number of pure strategies and each component x_{ih} denotes the probability that player i choose its hth pure strategy. The strategy space lies on the m-dimensional standard simplex Δ_m where: $\sum_{h=1}^{m} x_{ih} = 1$ and $x_{ih} \geq 0$ for all h. The expected payoff of a pure strategy e^h in a single game is $u(e^h, x) = e^h \cdot Ax$ where A is the $m \times m$ payoff matrix. The average payoff of all the player strategies is $u(x, x) = \sum_{h \in S} x_h u(e^h, x)$. In order to find the Nash equilibria of the game it is used the replicator dynamic equation (Taylor and Jonker, 1978)

$$\dot{x} = [u(e^h, x) - u(x, x)] \cdot x^h \; \forall h \in S \qquad (1)$$

which allows better than average strategies (best replies) to grow. As in (Erdem and Pelillo, 2012) we used the discrete time version of the replicator dynamic equation:

$$x^h(t+1) = x^h(t) \frac{u(e^h, x)}{u(x, x)} \; \forall h \in S \qquad (2)$$

where at each time step t the players update their strategies until the system converges and the Nash equilibria are found.

3 WSD Games

In this section we will show how we created the data necessary for our framework and how the games are played.

3.1 Graph Construction

We model the geometry of the data as a graph, with nodes corresponding to the words to be disambiguated, denoted by $I = \{i_j\}_{j=1}^{N}$, where i_j corresponds to the j-th word and N is the number of target words in a specific text. From I we construct a $N \times N$ similarity matrix W where each element w_{ij} is the similarity value assigned for the words i and j. W can be exploited as an useful tool for graph-based algorithms since it is treatable as weighted adjacency matrix of a weighted graph.

A crucial factor for the graph construction is the choice of the similarity measure, $sim(\cdot,\cdot) \rightarrow \mathbb{R}$ to weights the edges of the graph. For our experiments we used similarity measures which compute the strength of co-occurrence between any two words i_i and i_j

$$w_{ij} = sim(i_i, i_j)\ \forall i, j \in I : i \neq j \qquad (3)$$

Specifically we used the modified Dice coheffi-cient ($mDice$) (Dice, 1945), the pointwise mutual information (PMI) (Church and Hanks, 1990) and the log likelihood ratio (D^2) (Dunning, 1993) These measure have been calculate using the Google Web1T corpus (Brants and Franz, 2006), a large collection of n-grams (with a window of max 5 words) occurring in one terabyte of Web documents as collected by Google.

At this point we have the similarity graph W, we recall that we will use this matrix in order to allow the words to play the games only with similar words. The higher the similarity among two words, the higher the reciprocal influence and the possibility that they belong to a similar class. For this reason, at first we smooth the data in W and then choose only the most significant js for each $j \in W$. The first point is solved using a gaussian kernel on W, $w_{ij} = \exp(-\frac{w_{ij}^2}{2\sigma^2})$, where σ is the kernel width parameter; the second point is solved applying a $k - nearest\ neighbor$ algorithm to W, which allows us to remove the edges which are less significant for each $i \in I$. In our experiments we used $\sigma = 0.5$ and $k = 25$. Moreover, this operation reduces the computational cost of the algorithm, which will focus only on relevant similarities.

3.2 The Strategy Space

In order to create the strategy space of the game, we first use WordNet (Mallery, 1995) to collect the sense inventories $M_i = 1, \ldots, m$ of each word, where m is the number of synsets associated to word i. Then we set all the sense inventories and obtain the list of all possible senses, $C = 1, \ldots, c$.

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

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

where each row corresponds to the strategy space of a player and each column corresponds to a sense. Formally it is a c-dimensional space Δ_c and each mixed strategy profile lives in the mixed strategy space of the game, given by the Cartesian product $\Theta = \times_{i \in I} \Delta_i$.

At this point the strategy space can be initialized with the following formula in order to follow the constraints described in Section 2.1

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

for all $i \in I$ and $j \in S$.

3.3 The Payoff Matrix

We encoded the payoff matrix of a WSD game as a sense similarity matrix among all the senses in the strategy spaces of the game. In this way the higher the similarity among two sense candidates, the higher the incentive for a player to chose that sense, and play the strategy associated to it.

The $c \times c$ sense similarity matrix Z is defined as follows:

$$z_{ij} = ssim(s_i, s_j)\ \forall i, j \in C : i \neq j \qquad (5)$$

In our experiments we used the $GlossVector$ measure (Patwardhan and Pedersen, 2006) in order to compute the semantic relatedness $ssim(\cdot,\cdot)$. This measure calculates the cosine similarity among two second order context vectors. Each vector is obtained from a WordNet super-glosse, which is the gloss of a synset plus the glosses of the synsets related to it.

run	sim	P	R	F1	math	med.	gen.
1	PMI	57.4	48.9	52.8	47.4	56.3	53.5
2	$mDice$	58.8	50.0	54.1	48.5	58.4	53.5
3	D^2	53.5	45.4	49.1	43.4	54.4	46.7

Table 2: The results of the WSD-games team at SemEval-2015 task 13. Precision, Recall and F1 in all domains and F1 in specific domains.

From Z we can obtain the partial semantic similarity matrix for each pair of player, $Z_{ij} = m \times n$, where m and n are the senses of i and j in Z.

In a previous work (Tripodi et al., 2015) we did not use this information, instead we used labeled data points to propagate the class membership information over the graph. In this new version the use of the semantic information made the algorithm completely unsupervised.

3.4 System Dynamics

Now that we have the topology of the data W, the strategy space of the game S and the payoff matrix Z we can compute the Nash equilibria of the game according to equation (2). So in each iteration of the system each player gain its payoffs according to equation (6) which allows each payoff to be proportional to the similarity (w_{ij}) and to the affinity that player j has to the hs strategy of player i.

$$u_i(e^h, x) = \sum_{j \in N_i} ((w_{ij} Z_{ij}) x_j)_h \qquad (6)$$

When the system converges each player chooses the strategy with the highest value.

4 Results and Analysis

The dataset proposed by the organizers of SemEval-2015 Task 13 (Moro and Navigli, 2015) consists of five texts from three different domains: math and computer, biomedical and general. The english corpus is composed of 1426 instances to disambiguate and 1262 of them have been used in the evaluation. For our experiments we used only the instances whose lemma has an entry in WordNet 3.0 without looking up multi-words or trying to link the entities to other sources such as Wikipedia or BabelNet (Navigli and Ponzetto, 2012)

We submitted three runs for our system with 1227 single words disambiguated for each run. The only difference for each run is the similarity measure that we used to construct the graph W. For run-1 we used the PMI measure, for run-2 the $mDice$ coefficient and for run-3 the D^2. As we expected from previous experiments on similar datasets, the best results have been achieved using the $mDice$ coefficient (see Table 2). We obtained low recall values for all our runs and this because we did not search multi-words and did not use other sources of information for the named entities, in fact the number of named entities is limited in WordNet.

Looking more closely at the results, we noticed that we obtained a very low precision (48.5%) in the math and computer domain and this because even if the lexical entry of certain instances (eg. in text2: *tab*, *dialog*, *script*) have an entry in WordNet, their intended meaning is not present; it can only be accessible to those systems which use BabelNet to collect the sense inventories. This unexpected problem affects the performances of the system because even if those instances will not be considered in the evaluation, they have been used by other instances in our system to play the disambiguation games, compromising the dynamics of the system.

5 Conclusions and Future Works

We have presented an unsupervised system for WSD based on EGT which takes into account contextual similarity and semantic similarity information in order to perform a consistent labeling of the data. Its performances are below those of supervised systems and are comparable with unsupervised and semi-supervised systems even if on the Semeval-2015 task 13 dataset we did not use other source of information except WordNet, did not search multi-words and did not aspect that the intended meaning of some instances is not present in WordNet.

As future work we are planning to do a detailed evaluation of the system in order to find the most appropriate measures to use and to incorporate in the framework other sources of information like BabelNet. Furthermore we are also thinking to test the system as supervised and semi-supervised, implementing a new initialization of the strategy space and to test new graph construction techniques.

References

Eneko Agirre, Oier Lopez De Lacalle, Aitor Soroa, and Informatika Fakultatea. 2009. Knowledge-based WSD and Specific Domains: Performing Better than Generic Supervised WSD. In *IJCAI*, pages 1501–1506.

Eneko Agirre, Oier Lopez de Lacalle, and Aitor Soroa. 2014. Random Walks for Knowledge-Based Word Sense Disambiguation. *Computational Linguistics*, 40(1):57–84.

Thorsten Brants and Alex Franz. 2006. {Web 1T 5-gram Version 1}.

Kenneth Ward Church and Patrick Hanks. 1990. Word Association Norms, Mutual Information, and Lexicography. *Computational linguistics*, 16(1):22–29.

Diego De Cao, Roberto Basili, Matteo Luciani, Francesco Mesiano, and Riccardo Rossi. 2010. Robust and Efficient PageRank for Word Sense Disambiguation. In *Proceedings of the 2010 Workshop on Graph-based Methods for Natural Language Processing*, pages 24–32.

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

Ted Dunning. 1993. Accurate Methods for the Statistics of Surprise and Coincidence. *Computational linguistics*, 19(1):61–74.

Aykut Erdem and Marcello Pelillo. 2012. Graph Transduction as a Noncooperative Game. *Neural Computation*, 24(3):700-723.

John R. Firth. 1957. A Synopsis of Linguistic Theory 1930-1955. *Studies in linguistic analysis. Oxford: Blackwell*.

Taher H. Haveliwala. 2002. Topic-Sensitive PageRank. In *Proceedings of the 11th international conference on World Wide Web*, pages 517–526.

Robert A. Hummel and Steven W. Zucker. 1983. On the Foundations of Relaxation Labeling Processes. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, (3):267–287.

Kevin Leyton-Brown and Yoav Shoham. 2008. Essentials of Game Theory: A Concise Multidisciplinary Introduction. *Synthesis Lectures on Artificial Intelligence and Machine Learning*, 2(1):1–88.

John C. Mallery. 1995. WordNet: a Lexical Database for English. *Communications of the ACM*, 38(11):39-41.

Diana McCarthy, Rob Koeling, Julie Weeds, and John Carroll. 2007. Unsupervised Acquisition of Predominant Word Senses. *Computational Linguistics*, 33(4):553–590.

Rada Mihalcea, Paul Tarau, and Elizabeth Figa. 2004. PageRank on Semantic Networks, with Application to Word Sense Disambiguation. In *Proceedings of the 20th international conference on Computational Linguistics*, page 1126.

Rada Mihalcea. 2005. Unsupervised Large-Vocabulary Word Sense Disambiguation with Graph-Based Algorithms for Sequence Data Labeling. In *Proceedings of the conference on Human Language Technology and Empirical Methods in Natural Language Processing*, pages 411–418.

Andrea Moro and Roberto Navigli. 2015. SemEval-2015 Task 13: Multilingual All-Words Sense Disambiguation and Entity Linking. In *Proceedings of SemEval-2015*.

Roberto Navigli and Simone Paolo Ponzetto. 2012. BabelNet: The Automatic Construction, Evaluation and Application of a Wide-Coverage Multilingual Semantic Network. *Artificial Intelligence*, 193:217–250.

Roberto Navigli. 2009. Word Sense Disambiguation: A Survey. *ACM Computing Surveys (CSUR)*, 41(2):10.

Martin A. Nowak, Natalia L. Komarova, and Partha Niyogi. 2001. Evolution of Universal Grammar. *Science*, 291(5501):114–118.

Lawrence Page, Sergey Brin, Rajeev Motwani, and Terry Winograd. 1999. The PageRank Citation Ranking: Bringing Order to the Web.

Siddharth Patwardhan and Ted Pedersen. 2006. Using WordNet-Based Context Vectors to Estimate the Semantic Relatedness of Concepts. In *Proceedings of the EACL 2006 Workshop Making Sense of Sense-Bringing Computational Linguistics and Psycholinguistics Together*, volume 1501, pages 1–8.

Marcello Pelillo. 1997. The Dynamics of Nonlinear Relaxation Labeling Processes. *Journal of Mathematical Imaging and Vision*, 7(4):309–323.

Thanh Phong Pham, Hwee Tou Ng, and Wee Sun Lee. 2005. Word Sense Disambiguation with Semi-Supervised Learning. In *Proceedings of the National Conference on Artificial Intelligence*, volume 20, page 1093.

Ahti-Veikko Pietarinen. 2007. *Game theory and linguistic meaning*.

Simone Paolo Ponzetto and Roberto Navigli. 2010. Knowledge-Rich Word Sense Disambiguation Rivaling Supervised Systems. In *Proceedings of the 48th annual meeting of the association for computational linguistics*, pages 1522–1531.

William H. Sandholm. 2010. *Population games and evolutionary dynamics*.

Ravi Som Sinha and Rada Mihalcea. 2007. Unsupervised Graph-based Word Sense Disambiguation Using Measures of Word Semantic Similarity. In *ICSC*, volume 7, pages 363–369.

Brian Skyrms. 2010. *Signals: Evolution, learning, and information*.

John M. Smith and George R. Price. 1973. The Logic of Animal Conflict. *Nature*, 246:15.

György Szabó and Gabor Fath. 2007. Evolutionary Games on Graphs. *Physics Reports*, 446(4):97-216.

Peter D. Taylor and Leo B. Jonker. 1978. Evolutionary Stable Strategies and Game Dynamics. *Mathematical biosciences*, 40(1):145–156.

Stephen Tratz, Antonio Sanfilippo, Michelle Gregory, Alan Chappell, Christian Posse, and Paul Whitney. 2007. PNNL: A Supervised Maximum Entropy Approach to Word Sense Disambiguation. In *Proceedings of the 4th International Workshop on Semantic Evaluations*, pages 264–267.

Rocco Tripodi, Marcello Pelillo, and Rodolfo Delmonte. 2015. An Evolutionary Game Theoretic Approach to Word Sense Disambiguation. In *Proceedings of NLPCS 2014*.

John Von Neumann and Oskar Morgenstern. 1944. *Theory of Games and Economic Behavior (60th Anniversary Commemorative Edition)*.

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

Jörgen W. Weibull. 1997. *Evolutionary game theory*.

Zhi Zhong and Hwee Tou Ng. 2010. It Makes Sense: A Wide-Coverage Word Sense Disambiguation System for Free Text. In *Proceedings of the ACL 2010 System Demonstrations*, pages 78–83.