Unsupervised Learning of Meaningful Semantic Classes for Entity Aggregates

Henry Anaya-Sánchez
NLP & IR Group, UNED
Juan del Rosal, 16
28040 Madrid, Spain
henry.anaya@lsi.uned.es

Anselmo Peñas
NLP & IR Group, UNED
Juan del Rosal, 16
28040 Madrid, Spain
anselmo@lsi.uned.es

Abstract

This paper addresses the task of semantic class learning by introducing a new methodology to identify the set of semantic classes underlying an aggregate of instances (i.e, a set of nominal phrases observed as a particular semantic role in a collection of text documents). The aim is to identify a set of semantically coherent (i.e., interpretable) and general enough classes capable of accurately describing the full extension that the set of instances is intended to represent. Thus, the set of learned classes is then used to devise a generative model for entity categorization tasks such as semantic class induction. The proposed methods are completely unsupervised and rely on an (unlabeled) open-domain collection of text documents used as the source of background knowledge.

We demonstrate our proposal on a collection of news stories. Specifically, we model the set of classes underlying the predicate arguments in a Proposition Store built from the news. The experiments carried out show significant improvements over a (baseline) generative model of entities based on latent classes that is defined by means of Hierarchical Dirichlet Processes.

1 Introduction

The problem of identifying semantic classes for words in Natural Language Processing (NLP) has been shown useful to address many text processing tasks, mainly in the context of supervised and semi-supervised learning, in which the development of systems suffers from data scarcity.

Although some semantic dictionaries and ontologies do exist such as WordNet (Miller, 1995) or DBPedia (Mendes et al., 2011), their coverage is rarely complete, especially for large open classes (e.g., very specific classes of people and objects), and they fail to integrate new knowledge. Thus, it helps a lot to firstly learn word categories or classes from a large amount of (unlabeled) training data and then to use these categories as features for the supervised tasks.

The general task of semantic class learning, which can be broadly defined as the task of learning classes of words and their instances from text corpora, has been addressed in a variety of forms that correspond to different application scenarios. Among these forms, we can find two that have been termed as *semantic class mining* (Shi et al., 2010; Huang and Riloff, 2010; Kozareva et al., 2008) and *semantic class induction* (Iosif et al., 2006; Grave et al., 2013). These have to do respectively with (i) the expansion of (seed) sets of instances labeled with class information (Knowledge Base population), and with (ii) automatic annotation of individual instances with their semantic classes in the context of a particular text.

In our research, we are more focused on the later. Specifically, we center on the task of providing a collection of instances with class information (what an entity is) in a given textual context, and then to eventually enrich the context with properties inherited from the semantic class.

Thus, an important issue addressed in our work is that of learning a set of *meaningful classes* to label a collection of instances composing a semantic aggregate. By *meaningful classes*, we refer to a set of classes showing the following two properties:

Proceedings of the 11th International Conference on Computational Semantics, pages 88–93,
London, UK, April 15-17 2015.

- be a general enough class so that it can represent other entity occurrences in similar contexts, but also
- be a specific enough and coherent class so that it directly reflects the most important entity properties that can be inherited from the textual context in which the instance occurs.

For example, in the context "*x1* throws a touchdown pass", entity *x1* should be assigned with classes entailing football *players* rather than just a generic class *person*, and more likely, receive the class *quarterback*.

With the term *semantic aggregate* we refer to a set of instances not completely chosen at random, but sharing some contextual relationships (e.g., a set of nominal phrases observed in a given syntactic/semantic relationship with a specific verb in a text corpus).

In this way, this paper proposes a new methodology to identify/learn the set of semantic classes underlying an aggregate of entity instances. The aim is to learn a set of semantically meaningful classes capable of accurately describing the full extension that the set of entity instances is intended to represent for a posterior application to semantic class induction.

Thus, we also go beyond and propose a generative model of instances based on the set of learned classes for the aggregates to allow the classification of individual occurrences of instances.

We evaluate our proposal from a collection of news. Specifically, we model the set of semantic classes underlying the predicate arguments in a Proposition Store (Peñas and Hovy, 2010) built from the news texts.

So far, it has said little about the quality of automatically learned classes beyond their coverage; which is traditionally measured when evaluating approaches in application scenarios related to the task of semantic class mining, for expanding seed sets of words. By taking advantage of the generative method proposed to model instances, we rely on a recently introduced coherence measure to evaluate the coherence of the learned classes to classify instances. Also, we measure the generalization of instances by means of the likelihood of generating held-aside data.

The experiments carried out show significant improvements over a (baseline) generative model of instances based on latent classes that is defined by means of Hierarchical Dirichlet Processes (HDP) (Teh et al., 2006).

2 Learning semantic classes

We propose to learn the set of semantic classes underlying a (finite) semantic aggregate of entities instances A from a global set of candidate classes $C_0 = \{c_1, \ldots, c_{|C_0|}\}$ by relying on a learned value of Pointwise Mutual Information (PMI) between each possible class $c \in C_0$ and a model $\{p(c|A)\}_{c \in C_0}$ of posterior probabilities of classes conditioned on the aggregate. All statistics are learnt from a background text corpus in which the instances in A occur.

Specifically, we define by $Dom(A) = \{c | c \in C_0 \wedge pmi(c, A) > \theta_0 \wedge p(c|A) > \theta'_0\}$ the set of semantic classes underlying A; where θ_0 and θ'_0 are minimum thresholds, and $pmi(c, A)$ is the PMI value between c and A. [1] This value is calculated as follows:

$$pmi(c, A) = \log\left(\frac{p(c, A)}{p(c)p(A)}\right) = \log\left(\frac{p(c|A)}{p(c)}\right) \quad (1)$$

where both the model component $p(c|A)$ and prior $p(c)$ are learned relying on the following parameters:

- a $|C_0| \times |C_0|$ stochastic matrix $T = \{p(c_i|c_j)\}_{1 \leq i \leq |C_0|, 1 \leq j \leq |C_0|}$ representing a statistical mapping between possible classes in C_0 ($\forall j \in \{1, \ldots, |C_0|\}$, $\sum_{i=1}^{|C_0|} p(c_i|c_j) = 1$),
- a $|C_0| \times |I|$ stochastic matrix $Q = \{p(c_j|e_k)\}_{1 \leq j \leq |C_0|, 1 \leq k \leq |I|}$ that represents the distribution of possible classes conditioned on instances in the corpus (not only instances appearing in A but all instances in the corpus), and

[1] In our experiments, we set up to 1 and 0.001 the values of parameters θ_0 and θ'_0 respectively.

- a stochastic column vector $P(A) = \langle p(e_1|A), \ldots, p(e_{|I|}|A) \rangle^\top$ representing the distribution of observed instances in A.

From the above elements, the model of posterior class probabilities is estimated as:

$$p(c_i|A) = (T \cdot Q \cdot P(A))_i \tag{2}$$

being Q and $P(A)$ learned from maximum likelihood estimations; whereas the stochastic mapping T is learned by relying on infinite Markov chains between candidate classes as follows:

$$p(c_i|c_j) = \left((1-\alpha)(I_{|C_0|\times|C_0|} - \alpha T_0)^{-1}\right)_{i,j} \tag{3}$$

where $I_{|C_0|\times|C_0|}$ is the $|C_0| \times |C_0|$ identity matrix, T_0 is an $|C_0| \times |C_0|$ matrix whose element $(T_0)_{i,j}$ is defined as:

$$p_0(c_i|c_j) = \sum_{e \in I} p(c_i|e)p(e|c_j) \tag{4}$$

and α is the probability of adding a new candidate class to the Markov chain being generated. The idea of applying Markov chains is to obtain a "semantic-transitive" smoothing of mapping p_0 to define the mapping between classes. Note that such a mapping in turn applies a smoothing to the model of class posteriors given by $Q \cdot P(A)$.

Priors $p(c_1), \ldots, p(c_{|C_0|})$ correspond to the stationary distribution of candidate classes learned from the Markov chains.

2.1 Candidate classes

Assuming that Proposition Stores can be easily built in a unsupervised manner from the text corpus used as background (for example, such as in (Peñas and Hovy, 2010) that include class-instances observations in the one hand and propositions in the other), the process of obtaining candidate classes and class-instance counts to estimate the above models is straightforward.

Nevertheless, we additionally consider as candidate classes all common nominal phrases that do not appear as classes in the class-instance relation, but that appear as argument of propositions. These classes are considered to be singletons, whose entity instances are observed each time they occur as the argument of a proposition. Besides, we consider each (common) nominal phrase as instance of its head noun.

3 A generative model of instances

We consider a Probabilistic Topic Modeling (PTM) approach to model the intended extension of the classes underlying an aggregate as a realization from a generative model based on the learned classes.

Thus, similar to traditional PTM approaches such as LDA, we model each aggregate of instances $A = e_1, \ldots, e_{N_A}$ as a mixture:

$$p\left(e|A\right) = \sum_{i=1}^{k} p\left(e|c_{A_i}\right) p(c_{A_i}) \tag{5}$$

where e is an arbitrary instance, for all $i \in \{1, \ldots, k\}$, c_{A_i} is a class, $p(c_{A_i})$ is the prior for c_{A_i} in the generation of instances for A ($\langle p(c_{A_1}), \ldots, p(c_{A_k}) \rangle \sim \text{Dirichlet}_k(\alpha_C)$), and $p\left(e|c_{A_i}\right)$ is the probability of instantiating class c_{A_i} with instance e.

However, different from traditional PTM approaches, classes in Equation 5 do not correspond to true latent distributions of instances. Instead, each class in the mixture is actually a class in $Dom(A)$.

The probability value $p\left(e|c_{A_i}\right)$ in learning time is estimated from the parameters learned in the previous section to set up the model of class posteriors conditioned on the individual instances; whereas for applying the learned model to instance classification, the value is estimated from the counts of class assignments to entities in the text corpora in a similar way as LDA does (i.e., by applying a Dirichlet smoothing to the distribution of instances labeled with class c_{A_i} when learning the model). The aim is to allow applying the model to unseen instances for semantic class induction, while respecting the class-instance distribution learned in previous section.

Table 1: Averaged values of UMass coherence obtained (using ϵ=1.0e-50) for the clustering-based distributions of instances induced by the generative models.

Method	n=5	n=10	n=15	n=20
HDP	-217.051	-1271.62	-3665.42	-8073.6
Our proposal	-1.9693	-8.3945	-18.8997	-33.4624

4 Experiments

In order to evaluate our proposal, we consider a collection of 30,826 New York Times articles about US football, from which we build two proposition stores: one for training (based on the first 80% of the published articles) and the other one for testing (based on the remainder articles). The aim was to learn the semantic classes underlying each predicate argument taken as semantic aggregate of instances.

Specifically, documents in the training set were parsed using a standard dependency parser De Marneffe and Manning (2008); Klein and Manning (2003) together with TARSQI Verhagen et al. (2005), and after collapsing some syntactic dependencies following Clark and Harrison (2009); Peñas and Hovy (2010), we select the collection of 1,646,583 propositions corresponding to the top 1500 more frequent verb-based predicates (i.e., about the 90 percent of the total number of propositions in the training) to set up the input proposition store. The same procedure was applied to gather propositions from the test set, but they were held-aside for testing purposes.

We applied our approach to learn the classes underlying each predicate argument from the proposition store used for training, and evaluate the obtained models by conducting two experiments. In each experiment, we choose to compare the results obtained by our proposal to a baseline produced by applying HDP Teh et al. (2006) to infer latent distributions of distributions of instances, instead of using the PTM approach described in Section 3. [2]

4.1 Evaluating coherence

Thus, the first experiment was aimed at measuring the coherence or degree of interpretability of each distribution of instances induced from the class-based generative models. For this purpose, we rely on the UMass measure of coherence as defined in Stevens et al. (2012), that in this case was defined by regarding the co-occurrence frequencies of instances across the predicate arguments.

The UMass measure of coherence is intrinsic in nature. Significantly, it computes its counts from the training corpus used to train the models rather than a test corpus Stevens et al. (2012). So that, it attempts to confirm that the models learned data known to be in the corpus. This measure has been shown to be in agreement with coherence judgments by experts Mimno et al. (2011) in PTM.

In Table 1, we show the averaged values of UMass coherence obtained by each approach. As can be seen, the greatest values of the coherence measure correspond to the distributions of instances underlying the classes learned by our approach. This directly corroborates the good performance of the proposed model to learn coherent classes of entities to semantically label the aggregates of instances. In all cases, HDP significantly performs the worst in this experiment. To illustrate how the obtained values of coherence are representative enough of actual coherent distributions of instances, we show in Table 2 the classes learned for some predicate arguments. As can be seen, our approach accurately capture the more likely meaning of each argument.

4.2 Evaluating the generalization performance

The second experiment was focused on evaluating the generalization performance of our proposal, which we measure in terms of the generation of the held-aside argument instances in the test set. The average

[2]HDP is a fully bayesian, unsupervised PTM approach that differently from LDA and related (traditional) PTM approaches does not need to known the number of topics (in our case, instance distributions) to be discovered beforehand. Besides, HDP has been shown to optimize the generative approach of LDA in terms of the likelihood of predicting data.

Table 2: Examples of the classes learned by our approach for some predicate arguments.

Predicate	Arg.	Classes learned
win	*x* win -	team, group, no., person, champion,[football,team], host, giants, defeat, 49er, opponent, [defend,champion], victory,[super bowl,champion], [only,team],[other,team]
win	- win *y*	game,championship,title,[national,championship], [first,game], [last,game],[football,game],bowl,victory,job,[playoff,game],[straight,game],award, [consecutive,game], one,[division,title],division,[final,game],[home,game],[big,game], [championship,game],[run,game],[super,bowl],[regular-season,game],[national,title],battle
pass	*x* pass -	touchdown, group, person, [first,touchdown]
pass	- pass *y*	yard, season, test, play, touchdown, record, interception, ball, situation, examination, physical, [big,play], attempt, efficiency, rush, mark, downs, [last,season], yardage, offense, completion, [total,yard], more, rusher, person, protection, one
make	*x* make -	team, group, person, that, kicker
make	- make *y*	play, decision, mistake,[big,play],catch, playoff, start, change, move, difference, appearance, call, offer, deal, choice, money, debut, sense, statement,score, progress,trip, one, interception

Table 3: Averaged values and standard deviation of log-likelihood on the generation of instances for predicate arguments in the test data.

Method	Avg. likelihood	Std. dev.
HDP	-2009.77	169.52
Our approach	-1638.55	95.0238

log-likelihood on the generation of instances from each predicate argument was adopted to measure the performance in this case.

Table 3 summarizes the results obtained in this experiment. As it is shown, our approach, again, largely outperforms the approach based on HDP. All these results suggest that the classes learned by our approach can be properly applied to perform the identification of semantic classes, specially to address the task of semantic class induction.

5 Conclusions

In this paper, a new methodology to identify the set of semantic classes underlying an aggregate of instances (namely, a set of nominal phrases observed as predicate arguments in a collection of text documents) has been introduced. The aim of the methodology is to identify a set of semantically coherent (i.e., interpretable) and general classes capable of accurately describing the full extension that the set of instances is intended to represent. The set of learned classes was then used to devise a generative model for entity categorization tasks such as semantic class induction. The experiments carried out over a collection of news show significant improvements over a (baseline) generative model of instances (in terms of coherence and generalization) that is based on latent classes defined by means of Hierarchical Dirichlet Processes. Future work includes the application of our proposal to model generalized propositions as tuples of classes to directly address the task of semantic class induction.

Acknowledgments

This work was partially funded by MINECO (PCIN-2013-002-C02-01) and EPSRC (EP/K017845/1) in the framework of CHIST-ERA READERS project, and by project Voxpopuli (TIN2013-47090-C3-1).

References

Clark, P. and P. Harrison (2009). Large-scale extraction and use of knowledge from text. In *Proceedings of the fifth international conference on Knowledge capture*, pp. 153–160. ACM.

De Marneffe, M.-C. and C. D. Manning (2008). The stanford typed dependencies representation. In *Coling 2008: Proceedings of the workshop on Cross-Framework and Cross-Domain Parser Evaluation*, pp. 1–8.

Grave, E., G. Obozinski, F. Bach, et al. (2013). Hidden markov tree models for semantic class induction. In *CoNLL-Seventeenth Conference on Computational Natural Language Learning*.

Huang, R. and E. Riloff (2010). Inducing domain-specific semantic class taggers from (almost) nothing. In *Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics*, pp. 275–285. Association for Computational Linguistics.

Iosif, E., A. Tegos, A. Pangos, E. Fosler-Lussier, and A. Potamianos (2006). Unsupervised combination of metrics for semantic class induction. In *Spoken Language Technology Workshop, 2006. IEEE*, pp. 86–89. IEEE.

Klein, D. and C. D. Manning (2003). Accurate unlexicalized parsing. In *Proceedings of the 41st Annual Meeting on Association for Computational Linguistics-Volume 1*, pp. 423–430.

Kozareva, Z., E. Riloff, and E. H. Hovy (2008). Semantic class learning from the web with hyponym pattern linkage graphs. In *Proceeding of the ACL*, Volume 8, pp. 1048–1056.

Mendes, P. N., M. Jakob, A. García-Silva, and C. Bizer (2011). Dbpedia spotlight: shedding light on the web of documents. In *Proceedings of the 7th International Conference on Semantic Systems*, pp. 1–8. ACM.

Miller, G. (1995). Wordnet: A lexical database for english. *Communications of the ACM 38*(11), 39–41.

Mimno, D., H. Wallach, E. Talley, M. Leenders, and A. McCallum (2011). Optimizing semantic coherence in topic models. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, pp. 262–272.

Peñas, A. and E. Hovy (2010). Filling knowledge gaps in text for machine reading. In *Proceedings of the 23rd International Conference on Computational Linguistics: Posters*, pp. 979–987. Association for Computational Linguistics.

Shi, S., H. Zhang, X. Yuan, and J.-R. Wen (2010). Corpus-based semantic class mining: distributional vs. pattern-based approaches. In *Proceedings of the 23rd International Conference on Computational Linguistics*, pp. 993–1001. Association for Computational Linguistics.

Stevens, K., P. Kegelmeyer, D. Andrzejewski, and D. Buttler (2012). Exploring topic coherence over many models and many topics. In *Proceedings of the 2012 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning*, pp. 952–961. Association for Computational Linguistics.

Teh, Y., M. Jordan, M. Beal, and D. Blei (2006). Hierarchical dirichlet processes. *Journal of the American Statistical Association 101*(476), 1566–1581.

Verhagen, M., I. Mani, R. Sauri, R. Knippen, S. B. Jang, J. Littman, A. Rumshisky, J. Phillips, and J. Pustejovsky (2005). Automating temporal annotation with tarsqi. In *Proceedings of the ACL 2005 on Interactive poster and demonstration sessions*, pp. 81–84.