Combining Feature-Based and Instance-Based Transfer Learning Approaches for Cross-Domain Hedge Detection with Multiple Sources

Huiwei Zhou(✉), Huan Yang, Long Chen, Zhenwei Liu, Jianjun Ma, and Degen Huang

School of Computer Science and Technology, Dalian University of Technology, Dalian 116024, Liaoning, China
{zhouhuiwei, majian, huangdg}@dlut.edu.cn,
{yanghuan_dlut, chenlong.415, liuzhenwei}@mail.dlut.edu.cn

Abstract. The difference of hedge cue distributions in various domains makes the domain-specific detectors difficult to extend to other domains. To make full use of out-of-domain data to adapt to a new domain and minimize annotation costs, we propose a novel cross-domain hedge detection approach called FIMultiSource, which combines instance-based and feature-based transfer learning approaches to make full use of multiple sources. Experiments carried on BioScope, WikiWeasel, and FactBank corpora show that our approach works well for cross-domain uncertainty recognition and always improves the detection performance compared to other state-of-the-art instance-based and feature-based transfer learning approaches.

Keywords: Hedge detection · Cross-domain · Transfer learning

1 Introduction

Hedge detection has attracted more and more interest in natural language processing (NLP) community. The CoNLL-2010 Shared Task [1] addresses the detection of uncertainty in two domains (biological publications and Wikipedia articles). Hedging is widely used in various domains from science to humanities [2]. To date, several corpora annotated for uncertainty are publicly available in different domains, such as BioScope [3] from biomedical domain, FactBank [4] from newswire domain, WikiWeasel [1] from encyclopedia domain, etc. The uncertainty cue vocabulary and the distribution of certain and uncertain senses of cues vary in different domains [2].

The currently existing approaches focus on learning the domain-specific detectors in each publicly available corpus. Such approaches works well in the domains in which the training data are sufficient. Though the several publicly available corpora cover different aspect of uncertainty, the domain dependency of hedge detection makes the model trained on existing corpora does not perform well in other domains. It is impractical to annotate training data for each of the application domains. Szarvas et al. [2] show that

X. Zhang et al. (Eds.): SMP 2015, CCIS 568, pp. 225–232, 2015.
DOI: 10.1007/978-981-10-0080-5_22

distant domain data can contribute to the recognition of uncertainty cues, which efficiently reduces the annotation costs for a new domain. However, the target training data Szarvas et al. [2] need are 1,000 $\sim$ 2,000 labeled sentences, which are also too expensive.

This paper addresses the problem of detecting hedge in the case that the training data in the target domain are rarely scarce, for example, 200 instances. We propose a robust cross-domain hedge detection method, which combines feature-based transfer learning (or feature-transfer) and instance-based transfer learning (or instance-transfer) approaches to transfer hedge detection knowledge from multiple sources. The proposed method first trains a feature-transfer classifier and an instance-transfer classifier by using each source data individually in combination with the target data. The final ensemble classifier is given by a sum of the individual classifiers' predictions to overcome the weakness of a single transfer approach from a single source. The experiments show that our method outperforms the state-of-the-art transfer learning approaches in hedge detection, especially for a very small insufficient training set.

2 Related Work

The early hedge detection systems use handcrafted cue lexicons for cue recognition [5]. However, not all occurrences of the cues indicate uncertainty. With the development of publicly available corpora, the hedge detection task is treated as a sequential labeling [6, 7] or a token classification problem [8]. The former predicts for all tokens whether a token is inside a cue or outside a cue. The latter matches cue candidates based on a cue lexicon, and then classifies whether they denote uncertainty. All of these studies focus on in-domain cue detection, which assume that the training data are sufficient to get an acceptable model in the application domain. However, there are only several available corpora constructed for special domains, and the set of cues used and frequency of their certain and uncertain usages are domain dependent.

In the case that the training and working data follow a different data distribution, transfer learning could improve the performance without much expensive data-annotation effort. The existing transfer learning approaches can be divided into two cases: feature-transfer learning, and instance-transfer learning approaches. Feature-transfer learning [9, 10] tries to discovery latent common features shared by the source and target domains. Daumé III [9] proposes a Frustratingly Easy Domain Adaptation approach (Hereafter, FruDA for short), which augments the feature space of both the source and target data and uses the result as the input to a standard learning algorithm.

Instance-transfer learning [11–13] attempts to select the important source instances which are reused in the target domain by reweighting. TrAdaBoost [11] focuses on using a large set of labeled source data and a small set of labeled target data to automatically adjust the weights of training instances by Boosting. However, the limited labeled target data cannot represent the whole target domain sufficiently. Active learning algorithms are usually used to choose the target instances for representing the target distribution [12]. On the other hand, TPTSVM [13] uses semi-supervised learning to improve the generalization performance.

Transfer learning relying on one source could lead to negative transfer. Yao and Doretto [14] propose the MutiSourceTrAdaBoost algorithm to import knowledge from multiple sources to decrease the risk of negative transfer. Eaton and desJardins [15] propose a TransferBoost algorithm, which increases the weight of sources that show positive transfer to the target. Dredze et al. [16] adapt multiple source domain classifiers to a new target domain. However, they train domain-specific classifiers individually in each source domain without exploiting a small target training data set.

To take advantages of both feature-transfer learning and instance-transfer learning approaches and import knowledge from multiple sources, this paper proposes a method to integrate the two approaches for cross-domain hedge detection with multiple sources (FIMultiSource). The proposed method first trains a feature-transfer classifier and an instance-transfer classifier by using each source domain data set and a small number of target domain data set. Finally, the classifiers trained in different approaches with different sources are combined to take advantage of their various strengths.

3 Cross-Domain Hedge Detection

3.1 Hedge Cue Detection

Hedge information almost presents by lexical cues. However, the words in lexicon do not always present uncertainty but according to the context. This paper formulates the cue detection problem as a classification problem of candidate cues in lexicon. How to classify the candidate cues by using the limited labeled data in target domain to leverage multiple sources is the main task we want to solve in this paper. Several basic features, which are provided by GENIA Tagger, are used in our system.

(1) Current token features: $Token(i)$ $(i = 0)$
(2) Stem features: $Stem(i)$ $(i = -1, 0, +1)$
(3) POS features: $POS(i)$ $(i = -1, 0, +1)$
(4) Chunk features: $Chunk(i)$ $(i = -1, 0, +1)$
(5) Co-occurrence features: $Co(i)$ $(i = -1, 0, +1)$.

Other candidate cues that occur in the same sentence. Where $Co(-1)$ is the first candidate cue to the left, $Co(+1)$ is the first candidate cue to the right.

3.2 Transfer Learning with Multiple Sources

Given three kinds of data sets, a very small target training set $D_l = \{(x_i^l, y_i^l)|i = 1, \ldots, n\}$, $y_i^l = \{-1, +1\}$, the target domain unlabeled set $D_u = \{(x_j^u)|j = 1, \ldots, m\}$, and multiple source domain labeled sets $D_{s1}, \ldots, D_{st}, \ldots, D_{sq}$, where $D_{st} = \{(x_k^{st}, y_k^{st})|k = 1, \ldots, |st|\}$, $y_k^{st} = \{-1, +1\}$. n, m and $|st|$ are the sizes of D_l, D_u and D_{st}. Assume that D_l and D_u are in the same domain with same distribution, but D_l is not sufficient to training a model to classify the D_u. Our transfer learning algorithm tries to use D_{st} and D_u to help the insufficient training set D_l to train a better classifier $F(x_j)$ that minimizes the prediction error on the target unlabeled set D_u.

We combine instance-transfer and feature-transfer learning approaches to make full use of multiple sources. A formal description of FIMultiSource is given in Algorithm 1. For each source data D_{st}, FIMultiSource trains a feature-transfer classifier on $D_{st} \cup D_l$ and an instance-transfer classifier on $D_{st} \cup D_l \cup D_u$(or $D_{st} \cup D_l$). The final ensemble classifier is given by a sum of the individual classifiers' predictions to overcome the weakness of a single transfer approach from a single source.

Algorithm 1.

Input the source domains labeled data sets $D_{s1}, \ldots, D_{st}, \ldots, D_{sq}$, the target domain labeled data set D_l, the unlabeled target data set D_u, and the number of iteration N.

For $t = 1, \ldots, |sq|$

1. Train a feature-transfer classifier $Feature_t(x)$ on $D_{st} \bigcup D_l$.
2. Train an instance-transfer classifier $Instance_t(x)$ on $D_{st} \bigcup D_l \bigcup D_u$ (or $D_{st} \bigcup D_l$), for $d = 1, \ldots, N$.

end for

Output the final ensemble classifier $F(x_j) = sign(\sum_{t=1}^{|sq|} Feature_t(x_j) + Instance_t(x_j))$.

For feature-transfer, FruDA [9] is adopted for our FIMultiSource method. FruDA [9] projects the source data and target data $X = \mathrm{R}^F$ to an augmented space $\tilde{X} = \mathrm{R}^{3F}$ by the mappings Φ^s and Φ^t respectively:

$$\Phi^s(x) = <x, x, 0>, \ \Phi^t(x) = <x, 0, x> \tag{1}$$

where, $0 = <0, \ldots, 0> \in \mathrm{R}^F$ is the zero vector, F is the dimension of the initial input. Augment reduces difference between a source and a target domain by mapping the source data and target data into a common space. However, the scarce training data in the target domain is insufficient for represent the feature distribution of target domain.

For instance-transfer, TrAdaboost [11] and TPTSVM [13] are used for our FIMultiSource method respectively. Comparing with TrAdaboost, TPTSVM queries the most confident target domain unlabeled data D_u into training data, which helps to overcome the over-fitting of TrAdaBoost.

4 Experiments

4.1 Performance Based on Transfer Learning with One Source

We evaluate our method on three corpora: BioScope [3] from biomedical domain, WikiWeasel [1] from encyclopedia domain, and FactBank [4] from newswire domain. The BioScope corpus contains clinical texts as well as biomedical texts from full paper and scientific abstracts. Only scientific abstract is used as biomedical data in our experiments. The number of cues in biomedical, encyclopedia and news domains is

2694, 3265 and 720 respectively. The number of candidate cues in the three domains is 10272, 17009 and 2758 respectively. The training data in the target domain used in the experiments are only 200 instances.

We first evaluate our feature-transfer and instance-transfer learning combination approach in hedge detection by using only one source domain. Our transfer learning combination method is compared with the following basic methods:

Sou: This method applies the SVM with only the current source training set D_{st}.

Tar: This method applies the SVM with only the target training set D_l.

Sou + Tar: This method applies the SVM with both the source and target training sets $D_{st} \cup D_l$.

Fru: This method applies the feature-transfer learning approach FruDA with both the source and target training sets $D_{st} \cup D_l$.

TrA: This method applies the instance-transfer learning approach TrAdaBoost with both the source and target training sets $D_{st} \cup D_l$.

TPT: This method applies the instance-transfer learning approach TPTSVM with the three data sets: the source training set, the target training set and the target unlabeled set $D_{st} \cup D_l \cup D_u$.

Note that the former three basic methods are the traditional learning methods, while the later three basic methods are the transfer learning methods. To illustrate the effects of our transfer learning combination method, the two instance-transfer approaches (**TrA** and **TPT**) are combined with the feature-transfer approach respectively:

Fru + TrA: This method combines the outputs of the basic methods **Fru** and **TrA**.

Fru + TPT: This method combines the outputs of the basic methods **Fru** and **TPT**.

We generate six groups of experiments using all three data sets. Iteration time N of both TrAdaBoost and TPTSVM is 100. All the results below are the average of 10 repeats by random. The performance for hedge detection is evaluated by the official tool of the CoNLL-2010 shared task. The evaluation for hedge detection is carried out on the cue-level. The cue-level scores are based on the exact match of cue phrases.

The comparison of cue-level F-scores is shown in Table 1. Seen from the table, the proposed feature-transfer and instance-transfer combination approach outperforms the

Table 1. Comparison of Cue-level F-scores with One Source (abc = scientific abstract in biomedical domain; enc = encyclopedia)

Domain		Method							
Target	Source	Sou	Tar	Sou + Tar	Fru	TrA	TPT	Fru + TrA	Fru + TPT
news	abs	51.13	36.83	55.57	63.96	64.17	65.53	65.47	**66.10**
news	enc	67.18	36.83	67.29	68.52	66.24	66.67	68.67	**68.92**
abs	news	71.93	72.17	**84.70**	81.37	83.50	84.25	83.51	84.22
abs	enc	79.40	72.17	80.76	82.27	82.42	82.68	**83.90**	83.80
enc	abs	61.29	28.99	62.27	66.25	70.31	70.84	70.67	**71.19**
enc	news	70.99	28.99	72.08	69.16	72.23	73.23	72.81	**73.97**
Average		66.99	46.00	70.45	71.92	73.14	73.87	74.17	**74.70**

basic methods generally. Among the six basic methods, the three transfer learning methods (**Fru**, **TrA**, and **TPT**) are obviously better than the three traditional learning methods (**Sou**, **Tar**, and **Sou + Tar**). Since the training data are not sufficient (200 instances), the performance of **Tar** is very poor. The average cue-level F-score is only 46.00 %. **Sou + Tar** could improve **Tar** with the help of the source data. For the transfer learning methods, **TPT** always improves the F-scores of **TrA** by using the target unlabeled data to help transfer learning. Overall, **Fru + TPT** gives the best cue-level F-score 74.70 % through the novel combination of feature-transfer and instance-transfer learning as well as the utilization of the target unlabeled data.

4.2 Performance Based on Transfer Learning with Multiple Sources

We then perform the experiments with multiple sources. Three groups of experiments are generated using the same three data sets. Our FIMultiSource method is compared with the following basic methods employing multiple sources:

MSou: This method applies the SVM with the union of multiple source training sets $D_{s1} \cup \ldots \cup D_{sq}$.

MSou + Tar: This method applies the SVM with the union of multiple source training sets and the target training set $D_{s1} \cup \ldots \cup D_{sq} \cup D_l$.

MFru: This method extends FruDA to multiple source domains with the multiple source training sets and the target training set $D_{s1} \cup \ldots \cup D_{sq} \cup D_l$. For $K = q + 1$ domains totally, FruDA is simply extended to multiple sources by expanding the feature space to $\mathrm{R}^{(K+1)F}$, where "+1" corresponds to the "general domain" [9].

MTPT: This method extends TPTSVM to multiple source domains with the union of multiple source training sets, the target training set and the target unlabeled set $D_{s1} \cup \ldots \cup D_{sq} \cup D_l \cup D_u$. Zhou et al. [13] introduces a source factor $e^{(\varepsilon_t - \varepsilon_s)}$ to increase or decrease the source weight based on whether the union of the source and target domain data shows positive or negative transfer to the target domain. ε_t is the prediction error of the classifier trained with the target sets. ε_s is the prediction error of the classifier trained with both the target and source sets. We simply extend TPTSVM to multiple sources by using the source factor $e^{(\varepsilon_t - \varepsilon_{st})}$ to adjust the weight of source domain set D_{st}.

ComFru: This method combines the prediction value of each **Fru**, which is learned by transferring knowledge from each source to the same target.

ComTPT: This method combines the prediction value of each **TPT**, which is learned by transferring knowledge from each source to the same target.

The comparison of cue-level F-scores is shown in Table 2. From the table we can see that the proposed **FIMultiSource** method outperforms all six basic methods over all groups. Among the six basic methods, **MSou** and **MSou + Tar** perform poorly because they simply employ traditional learning with the union of training data. The average cue-level F-score of **MFru** is 69.59 %, while that of **MTPT** is 72.38 %. **ComFru** and **ComTPT** which combine the results of individual classifier could perform better than **MFru** and **MTPT**. The average cue-level F-scores of **MFru** and

Table 2. Comparison of Cue-level F-scores with Multiple Sources

Domain		Method						
Target	Source	MSou	MSou + Tar	MFru	MTPT	EnsFru	EnsTPT	FIMultiSource
news	abs, enc	55.00	57.16	62.18	64.35	68.27	67.71	**69.23**
abs	enc, news	78.94	80.14	81.48	82.09	82.93	84.75	**84.96**
enc	abs, news	61.16	61.78	65.10	70.71	69.14	73.54	**73.61**
Average		65.03	66.36	69.59	72.38	73.45	75.33	**75.93**

MTPT are even lower than that of **Fru** and **TPT** (see Table 1). **MFru** and **MTPT** adapt multiple sources to a new target domain in the transfer learning process. However, they do not get the results as good as expected. This is perhaps because transferring hedge detection knowledge from multiple sources to the target domain in one task is quite difficult. Our **FIMultiSource** achieves the cue-level F-score of 75.93 %, which is 1.23 % higher than that of **Fru + TPT**. The improvements benefit from multiple sources.

Szarvas et al. [2] uses the same three corpora (Bioscope, WikiWeasel, and FactBank). The target training set they need consists of 1,000 $\sim$ 2,000 sentences. The model trained by these sentences has already achieved the average F-score of 73.5 %. Szarvas et al. [2] adapt FruDA [9] to transfer hedge detection knowledge from one source to the target domain, and the average F-score reaches 77.8 %. We also employ FruDA in our experiments, and achieve the average F-score of 71.92 %. However, the target training data we need is rarely scarce (200 instances). By using our FIMultiSource method, the average F-score is improved to 75.93 %, which outperforms FruDA significantly.

5 Conclusion

In this paper, we proposed a novel approach FIMultiSource for cross-domain hedge detection with multiple sources. The proposed approach first learns cross-domain classifiers from each source to the target domain based on the feature-transfer and instance-transfer learning approach respectively. The final ensemble classifier is given by a sum of the individual classifiers' predictions to overcome the weakness of a single transfer approach from a single source. Experiments show that our FIMultiSource achieves the cue-level F-score of 75.93 %, which significantly outperforms previous state-of-the-art transfer learning approaches. Both feature-transfer and instance-transfer play an important role in cross-domain hedge detection. How to balance their function to optimize the combination of feature-transfer and instance-transfer for cross-domain hedge detection will be studied in our future work.

Acknowledgements. This research is supported by National Natural Science Foundation of China (Grant No. 61272375).

References

1. Farkas, R., Vincze, V., Móra, G., Csirik, J., Szarvas G.: The CoNLL-2010 shared task: learning to detect hedges and their scope in natural language text. In: Proceedings of the 14th Conference on Natural Language Learning (CoNLL-2010) Shared Task, pp. 1–12 (2010)
2. Szarvas, G., Vincze, V., Farkas, R., Móra, G., Gurevych, I.: Cross-genre and cross-domain detection of semantic uncertainty. Comput. Linguist. **38**, 335–367 (2012)
3. Vincze, V., Szarvas, G., Farkas, R., Móra, G., Csirik, J.: The BioScope corpus: biomedical texts annotated for uncertainty, negation and their scopes. BMC Bioinf. **9**, S9 (2008)
4. Saurí, R., Pustejovsky, J.: FactBank: A corpus annotated with event factuality. Lang. Resour. Eval. **43**, 227–268 (2009)
5. Light, M., Qiu, X.Y., Srinivasan, P.: The language of bioscience: facts, speculations, and statements in between. In: Proceedings of BioLink 2004 Workshop on Linking Biological Literature, Ontologies and Databases: Tools for Users, pp. 17–24 (2004)
6. Tang, B.Z., Wang, X.L., Wang, X., Yuan, B., Fan, S.X.: A cascade method for detecting hedges and their scope in natural language text. In: Proceedings of the 14th Conference on Computational Natural Language Learning: Shared Task, pp. 13–17 (2010)
7. Zhou, H.W., Li, X.Y., Huang, D.G., Yang, Y.S., Ren, F.J.: Voting based ensemble classifiers to detect hedges and theirs scopes in biomedical texts. IEICE Trans. Inf. Syst. **E94-D**(10), 1989–1997 (2011)
8. Velldal, E., Øvrelid, L., Oepen, S.: Resolving speculation: MaxEnt cue classification and dependency-based scope rules. In: Proceedings of the 14th Conference on Computational Natural Language Learning: Shared Task, pp. 48–55 (2010)
9. Daumé III, H.: Frustratingly easy domain adaptation. In: Proceedings of the 45th Annual Meeting of the Association of Computational Linguistics, pp. 256–263 (2007)
10. Zhang, H.N., Tian, Z., Kuang, R.: Transfer learning across cancers on DNA copy number variation analysis. In: Proceedings of the 13th International Conference on Data Mining (ICDM), pp. 1283–1288 (2013)
11. Dai, W.Y., Yang, Q., Xue, G.R., Yu, Y.: Boosting for transfer learning. In: Proceedings of the 24th International Conference on Machine learning, pp. 193–200 (2007)
12. Wang, X.Z., Huang, T.K., Schneider, J.: Active transfer learning under model shift. In: Proceedings of the 31st International Conference on Machine Learning (ICML-2014), pp. 1305–1313 (2014)
13. Zhou, H.W., Zhang, Y., Huang, D.G., Li, L.S.: Semi-supervised learning with transfer learning. In: Proceedings of the Chinese Computational Linguistics and Natural Language Processing Based on Naturally Annotated Big Data, pp. 109–119 (2013)
14. Yao, Y., Doretto, G.: Boosting for transfer learning with multiple sources. In: Proceedings of the 2010 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 1855–1862 (2010)
15. Eaton, E., desJardins, M.: Selective transfer between learning tasks using task-based boosting. In: Proceedings of the 25th AAAI Conference on Artificial Intelligence, pp. 337–342 (2011)
16. Dredze, M., Kulesza, A., Crammer, K.: Multi-domain learning by confidence-weighted parameter combination. Mach. Learn. **79**(1–2), 123–149 (2010)