Online Supplementary Materials: Genetic Programming for Document Classification: A Transductive Transfer Learning System

Wenlong Fu, Bing Xue, Xiaoying Gao, and Mengjie Zhang School of Engineering and Computer Science
Victoria University of Wellington, PO Box 600, Wellington 6140, New Zealand Emails: wenlong.fu@gmail.com,
{bing.xue,xiaoying.gao,mengjie.zhang}@ecs.vuw.ac.nz

TABLE I
TEST ACCURACIES (MEANS AND STANDARD DEVIATIONS) ON THE TARGET DOMAINS FROM GP_{SD}, GP_{GP}, $GP_{GP,2}$

data	GP_{SD}	GP_{GP}	$GP_{GP,2}$
$data_1$	0.655 ± 0.025	0.668 ± 0.025	0.639 ± 0.018
$data_2$	0.686 ± 0.030	0.707 ± 0.033	0.700 ± 0.021
$data_3$	0.634 ± 0.037	0.715 ± 0.051	0.715 ± 0.108
$data_4$	0.634 ± 0.036	0.679 ± 0.035	$0.698 \pm 0.037 \uparrow$
$data_5$	0.639 ± 0.036	0.712 ± 0.039	$0.739 \pm 0.036 \uparrow$
$data_6$	0.701 ± 0.038	0.742 ± 0.024	0.735 ± 0.011

I. TEST PERFORMANCES ON GP CLASSIFIERS ONLY

In order to check whether GP classifiers from GP_{GP} are used for further evolving GP classifiers in the "labelled" training data, the results from $GP_{GP,2}$ are shown in Table I. Here, $GP_{GP,2}$ uses the training data labelled by GP_{GP} to evolve new classifiers. Based on the multiple comparison results in Table I, means that the relevant results are the significantly worse than the two others, and ↑ for being the significantly best. There are two interesting observations. First, for $data_4$ and $data_5$ only, $GP_{GP,2}$ has the best test performances than GP_{SD} and GP_{GP}. there are not large performance improvements on the results from GP_{GP} to $GP_{GP,2}$. Second, the test results from $GP_{GP,2}$ are the worst among all the results on $data_1$. Therefore, it is not always to improve test performances when GP classifiers are evolved from the relabelled training data.

It is noted that Table II shows the SVM is better than GP to train classifiers on the target domain. It is possible that single GP classifiers are not more effective than SVM classifiers in the labelled training data. Single GP classifiers may be not sufficient to further improve more effectively labelled training data. It is a reason that SVM_{GP} is suggested for the transfer learning tasks, rather than GP_{GP}.

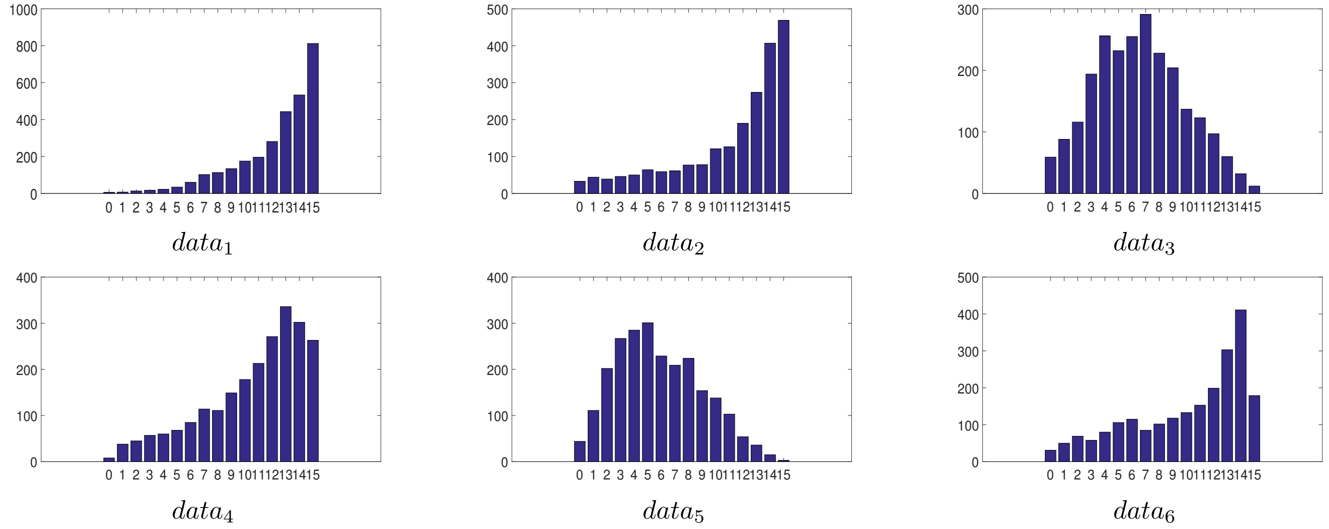
II. DETAILS OF DISTRIBUTIONS OF VOTING NUMBER OCCURRENCES FOR LABELS

The prediction diversity is very important for getting more correctly labelled training data, where rich diversity is needed for combining a set of classifiers to improve the test accuracy. This section investigates the accuracy improvement on labelling target training documents from the view of voting number occurrences.

This section provides more details of the distributions of voting number occurrences. The results in the main paper shows that there are obvious improvements when the linear SVM classifiers are retrained from the training documents labelled by the SVM classifiers from SVM_{GP}. The voting from GP classifiers on the source domain can be helpful to select proper training documents on the target domain for training linear SVM classifiers. However, the retrained linear SVM classifiers (voting from $SVM_{GP,2}$) cannot obviously further improve the test accuracies. In order to investigate these phenomena from SVM_{GP} and $SVM_{GP,2}$, Fig. 1 shows examples of the voting number distribution for one category on the target training documents using GP_{SD} classifiers. Note that all voting number distributions at each independent run on each dataset are similar to each other.

It is interesting that the voting numbers on $data_3$ and $data_5$ in Fig. 1 are located in middle of all possible voting numbers (from 0 to 15). There are few voting numbers with very high percentages of all voting numbers. However, the other four datasetes have high occurrences on the last few numbers (from 11 to 15). From the results on $V_{GP_{SD}}$ in Table **??**, the test accuracies on $data_3$ and $data_5$ are higher than the other four datasets. From a prediction point of view, various predictions from different GP classifiers will have voting numbers being most located around the middle range of all voting numbers. The prediction diversity of GP classifiers (not randomly guessing) makes contribution on the test performance improvement in the GP transfer learning system.

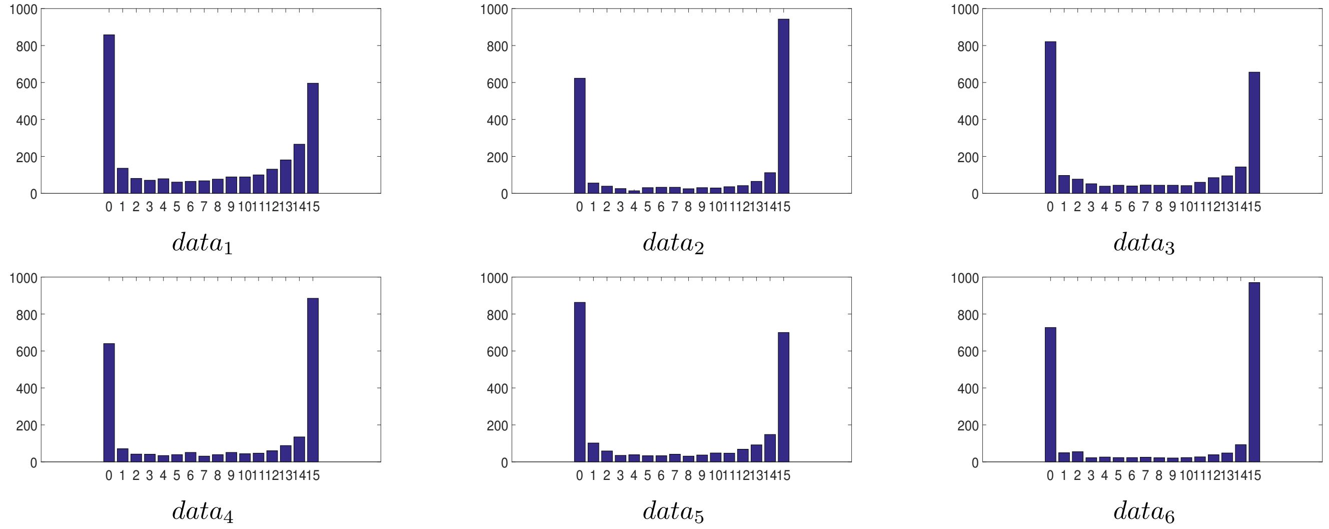
Fig.2 shows examples of voting number histograms for one category from SVM_{GP} on the six datasetse. Since SVM effectively trains classifiers on the six datasets using the training documents labelled by $V_{GP_{sd}}$, most of documents are predicted correctly, and the highest frequency voting numbers are 0 and 15 for one category. It looks like that the predictions from $V_{SVM_{GP}}$ are similar to each other and the retrained SVM classifiers ($SVM_{GP,2}$) have limited ability to be combined for test performance improvement. Note that it is easy to generate different SVM classifiers from $V_{SVM_{GP}}$, such as using a random subset of input features, or randomly sampling a subset of training documents. However, the accuracy of the trained classifiers should be kept. If the accuracy of labelling the training documents on these various classifiers are decreased, the voting accuracy on the training documents may be not good and the training documents labelled by these various classifiers are not sufficient for further training better classifiers. It is a challenging task to generate various



$data_1$ $data_2$ $data_3$

$data_4$ $data_5$ $data_6$

Fig. 1. Examples of Voting Number Occurrence Histograms for One Category on GP_{SD} classifiers.



$data_1$ $data_2$ $data_3$

$data_4$ $data_5$ $data_6$

Fig. 2. Examples of Voting Number Occurrence Histograms for One Category on SVM_{GP}.

classifiers while the accuracy of these classifiers are not decreased. This challenging task will be addressed in our future work.

Therefore, from the voting number occurrence distributions, when most of voting numbers are located at both terminal points, it is hard to get a further test performance improvement from the voting results. When most of the voting numbers are located in a wide range, the test performance can be further improved by the voting results. Based on the voting number occurrence distributions on the target domain, the GP transfer learning system can decide whether to retrain classifiers or not.

III. Performance Check on Supervised SVM and GP

Table II gives the test performances on SVM for the target test documents when SVM classifiers are trained from source training documents and target training documents separately. When learning from the SD training data is used to train SVM classifiers in Table II, SADA and FLDA are not used here. MGP evolves GP classifiers using the target training data with their labels. The test results from MGP are used to check the performance when the real labels of the target training data are provided. The results in Table II are different from the results of the proposed transfer learning system (without labels in the target training data). From the results, SVM can achieve high accuracy test performances on the six datasets (using subspaces) when SVM is directly trained from the target training datasets with labels. All relevant results are significantly better than results from MGP and GP_{SD}. However, when SVM classifiers are trained from source training datasets and are directly applied to test target documents, these SVM classifiers have the worse test performances, comparing with the others. Therefore, the datasets with the 30 dimensions can be used to train proper linear SVM classifiers. However, there are obvious differences of the transformed feature spaces between the source domain and the target domain. It is a

TABLE II
TEST PERFORMANCES (MEANS±STANDARD DEVIATIONS) OF SVM AND GP ON THE TARGET TEST DOCUMENTS.

data	Learned from TD training data		Learned from SD training data	
	SVM	MGP	SVM	GP_{SD}
$data_1$	$\mathbf{0.848 \pm 0.002}$	0.742 ± 0.017	0.333 ± 0.004	$\mathbf{0.659 \pm 0.026}$
$data_2$	$\mathbf{0.911 \pm 0.005}$	0.792 ± 0.024	0.555 ± 0.003	$\mathbf{0.683 \pm 0.027}$
$data_3$	$\mathbf{0.887 \pm 0.002}$	0.789 ± 0.039	0.426 ± 0.002	$\mathbf{0.624 \pm 0.039}$
$data_4$	$\mathbf{0.836 \pm 0.021}$	0.753 ± 0.036	0.389 ± 0.011	$\mathbf{0.636 \pm 0.043}$
$data_5$	$\mathbf{0.908 \pm 0.002}$	0.761 ± 0.035	0.497 ± 0.001	$\mathbf{0.634 \pm 0.038}$
$data_6$	$\mathbf{0.914 \pm 0.003}$	0.816 ± 0.026	0.581 ± 0.008	$\mathbf{0.703 \pm 0.031}$

potential reason that SADA and FLDA do not have good test performances on the six datasets.

IV. EXAMPLES OF GP CLASSIFIERS

$$f_{sd,1} = \frac{x[menu]}{\frac{14.12x[dumbest]x[dod]}{x[au]x[editor]}} \tag{1}$$

$$f_{sd,2} = 81.59x[brilliant](99.32+x[editor]+x[scratchy])+ 0.00013x[dumbest] - 3092.98 \tag{2}$$

$$f_{sd,3} = \frac{x[llnl]}{\frac{x[has]x[dumbest]}{x[research]x[window]}} \tag{3}$$

$$f_{sd,4} = \frac{x[dumbest]}{x[window]} - 1257.45 + 81.59\frac{x[dumbest]}{x[editor]} \tag{4}$$

$$tree_{sd,1} = x[dos] - \frac{f_{sd,1}}{f_{sd,2}} + x[program] - \frac{f_{sd,3}}{f_{sd,4}} \tag{5}$$

Eq. (5) provides an example of GP classifier $tree_{sd,1}$ from the source domain (GP_{SD}) on $data_5$. There are two interesting observations. First, there are four keywords for category "comp" , namely "dos", "menu", "program" and "windows". From Eq. (5), when the numbers of these four keywords are positive for discriminating a document to the category "comp". For instance, $x[dos] + x[program]$ is included in $tree_{sd,1}$. Since these four keywords are also included in the target domain, it is a reason why $tree_{sd,1}$ can classify a document from the target domain. It is interesting that $tree_{sd,1}$ mainly predicts a document as category "comp" or not. There are no very obvious keywords from the category "auto". Second, three general words "has", "research" and "editor" are included in $tree_{sd,1}$. From $tree_{sd,1}$, the influence from the three words are not obvious. It is noted that "dumbest" is used in "comp" with a very low frequency and the division is protected. Eq. (1) returns $x[menu]$ for most of predicted documents.

$$f_{td,1} = \begin{cases} x[yankees] & \text{if } x[mech] > x[paintbrush] \\ x[entirety] & \text{otherwise} \end{cases} \tag{6}$$

$$f_{td,2} = x[vendor]+x[microsoft]+x[imakefile]-62.55*f1 \tag{7}$$

$$f_{td,3} = (\frac{x[information]}{x[year]} - x[parrett])\frac{x[managed]}{x[sox]-x[honda]}) \tag{8}$$

$$f_{td,4} = \frac{x[punishments]}{x[baseball]x[baseball]} - x[purse] \tag{9}$$

$$f_{td,5} = \frac{x[bnr]}{3.22(x[mounted]+x[pitcdhing])} \tag{10}$$

$$f_{td,6} = x[hit] - x[volunteer] + x[add] - x[royals] \tag{11}$$

$$f_{td,7} = \frac{x[he]}{f_{td,3}f_{td,4}(f_{td,5}+f_{td,6})} \tag{12}$$

$$f_{td,8} = (\frac{x[adder]}{x[dealer]} - x[hit])(\frac{x[purse]}{x[car]} - x[mech]) \tag{13}$$

$$f_{td,9} = x[yankees]x[yankees] + x[entirety]) \tag{14}$$

$$f_{td,10} = f_{td,8} - \frac{57.54f_{td,9}}{x[share]x[yankees]x[yankees]} \tag{15}$$

$$f_{td,11} = \frac{x[paintbrush]}{x[transmission]} - x[ahead] - x[hrs] \tag{16}$$

$$f_{td,12} = \frac{x[year]}{x[kids]} - x[punishments] \tag{17}$$

$$f_{td,13} = \frac{x[ahead]}{x[purse]} - 0.04 * x[known] \tag{18}$$

$$f_{td,14} = x[bnr] - x[managed] + x[fan] \tag{19}$$

$$f_{td,15} = \begin{cases} f_{td,13} & \text{if } 0.04x[probe] > (x[ardent] - x[visited]) \\ f_{td,14} & \text{otherwise} \end{cases} \tag{20}$$

$$tree_{td,1} = \frac{f_{td,2}}{f_{td,7}f_{td,10}(f_{td,11}f_{td,12} - f_{td,15})} \tag{21}$$

Eq. (21) provides an example of GP classifier $tree_{td,1}$ from the target domain. There are four interesting observations. First, sub-equation $f_{td,1}$ (Eq. (6)) in $tree_{td,1}$ does not make an important contribution for classification. $f_{td,1}$ returns zero with a high frequency since there are few occurrences on the words "mech", "paintbrush" and "entirety" in $data_5$. When we analyse at $tree_{td,1}$, $f_{td,1}$ can be ignored. Second, keywords "microsoft" and "imakefile" are used for "comp", and keyword "yankees" for "rec.sport.baseball". Third, $f_{td,2}$ (Eq. (7)) plays very an import role to discriminate documents as category "rec". When there are no words "vendor", "microsoft" and "imakefile" in a predicted document, the document is marked as category "rec" with a very high chance. It is noted that $tree_{td,1}$ returns 1 when its divisor is zero. Four, when $f_{td,2}$ is larger than zero, it is only required to check whether the denominator of $tree_{td,1}$ is not larger than zero.

In summary, from $tree_{sd,1}$ and $tree_{td,1}$, it is possible to reasonably interpret GP classifiers. Some keywords shared by the source domain and the target domain are helpful to classify documents.