# Launch and Iterate: Reducing Prediction Churn Appendix

Q. Cormier<br/>ENS LyonM. Milani Fard, K. Canini, M. R. Gupta<br/>Google Inc.15 parvis René Descartes<br/>Lyon, France1600 Amphitheatre Parkway<br/>Mountain View, CA 94043quentin.cormier@ens-lyon.fr{mmilanifard, canini, mayagupta}@google.com

This appendix includes proofs of the theorems presented in the paper and details about the experimental design.

## A Proof of Theorem 1

*Proof.* Let z = (x, y) be a fixed testing sample. We define the function  $G_z : (\mathcal{X} \times \mathcal{Y})^{2m} \to \mathbb{R}$  as:

$$G_z(T, T') = \ell_\gamma(f_T(x), y) - \ell_\gamma(f_{T'}(x), y).$$
(1)

For any  $i \in \{1, ..., m\}$ , if we re-sample the *i*th element in T to get  $T^i$ , using the  $\beta$ -stability of the learning algorithm and Lipschitz continuity of  $\ell_{\gamma}$  we get:

$$|G_z(T,T') - G_z(T^i,T')| \le \frac{1}{\gamma} |f_T(x) - f_{T^i}(x)| \le \frac{\beta}{\gamma}.$$
(2)

The same inequality holds for  $|G_z(T,T') - G_z(T,T'^i)|$ . We have  $\mathbb{E}_{T,T'\sim D^m}[G_z(T,T')] = 0$ , and thus can apply the McDiarmid inequality to get:

$$\Pr_{T,T'\sim D^m}[|G_z(T,T')| > \epsilon] \le 2e^{-\frac{\epsilon^2 \gamma^2}{m\beta^2}}.$$
(3)

Integrating the above gives us the bound over the expectation:

$$\mathbb{E}_{T,T'\sim\mathcal{D}^m}[|G_z(T,T')|] \le \int_0^\infty \Pr_{T,T'\sim\mathcal{D}^m}[|G_z(T,T')| > \epsilon]d\epsilon \le \frac{\beta\sqrt{\pi m}}{\gamma}.$$
(4)

The above inequality holds for any fixed z and thus holds for the expectation:

$$\mathbb{E}_{T,T'\sim\mathcal{D}^m}[C_{\gamma}(f_1,f_2)] = \mathbb{E}_{T,T'\sim\mathcal{D}^m}\left[\mathbb{E}_{Z\sim\mathcal{D}}[|G_Z(T,T')|]\right]$$
(5)

$$= \mathbb{E}_{Z \sim \mathcal{D}} \left[ \mathbb{E}_{T, T' \sim \mathcal{D}^m} [|G_Z(T, T')|] \right]$$
(6)

$$\leq \frac{\beta\sqrt{\pi m}}{\gamma}.$$
(7)

## **B Proof of Theorem 2**

*Proof.* We again use the McDiarmid inequality, on the function  $H : (\mathcal{X} \times \mathcal{Y})^{2m} \to \mathbb{R}$  defined as:

$$H(T,T') = C_{\gamma}(f_T, f_{T'}) = \mathbb{E}_{(X,Y)\sim\mathcal{D}}[|\ell_{\gamma}(f_T(X), Y) - \ell_{\gamma}(f_{T'}(X), Y)|].$$
(8)

30th Conference on Neural Information Processing Systems (NIPS 2016), Barcelona, Spain.

For any  $i \in \{1, ..., m\}$ , if we re-sample the *i*th element in T to get  $T^i$ , using the  $\beta$ -stability of the learning algorithm and Lipschitz continuity of  $\ell_{\gamma}$  we get:

$$|H(T,T') - H(T^{i},T')| \leq \mathbb{E}_{(X,Y)\sim\mathcal{D}} \left[ ||\ell_{\gamma}(f_{T}(X),Y) - \ell_{\gamma}(f_{T'}(X),Y)| - (9) \right]$$

$$|\ell_{\gamma}(f_{T^{i}}(X), Y) - \ell_{\gamma}(f_{T'}(X), Y)||$$
(10)

$$\leq \underset{(X,Y)\sim\mathcal{D}}{\mathbb{E}}[|\ell_{\gamma}(f_{T}(X),Y) - \ell_{\gamma}(f_{T^{i}}(X),Y)|]$$
(11)

$$\leq \frac{\beta}{\gamma},$$
 (12)

where line (11) is by reverse triangular inequality. Same bound similarly holds for replacing the *i*th element in T':  $|H(T,T') - H(T,T'^i)| \le \beta/\gamma$ . Applying McDiarmid inequality and using the bound on the expectation of H from Theorem 1 completes the proof:

$$\Pr_{T,T'\sim\mathcal{D}^m}\left\{C_{\gamma}(f_T, f_{T'}) > \epsilon + \frac{\sqrt{\pi m}\beta}{\gamma}\right\} \leq \Pr_{T,T'\sim\mathcal{D}^m}\left\{H(T,T') > \epsilon + \mathbb{E}_{T,T'\sim\mathcal{D}^m}[H(T,T')]\right\}$$
$$\leq e^{-\frac{\epsilon^2\gamma^2}{m\beta^2}}.$$
(13)

# C Proof of Theorem 3

The proof partly follows Lemma 21 from [1] and Theorem 4 from [2]. Define:

$$\ell_j(g) = (g(x_j) - y_j)^2$$
(14)

$$\hat{R}_T(g) = \frac{1}{m} \sum_{j=1}^m \ell_j(g)$$
 (15)

$$\hat{R}_{T}^{\setminus i}(g) = \frac{1}{m} \sum_{\substack{j=1\\ j \neq i}}^{m} \ell_{j}(g)$$
(16)

$$R_T(g) = \hat{R}_T(g) + \lambda \|g\|_k^2$$
(17)

$$R_T^{\setminus i}(g) = \hat{R}_T^{\setminus i}(g) + \lambda \|g\|_k^2.$$
(18)

By the assumption of the theorem,  $f_T$  is the minimizer of  $R_T$ . Let  $f_T^{\setminus i}$  be the minimizer of  $R_T^{\setminus i}$ . Lemma 1. With the assumptions of Theorem 3, we have for all *i*:

$$\forall x : (f_T(x) - f_T^{\setminus i}(x))^2 \le \frac{\kappa^4}{\lambda^2 m^2} (f_T(x_i) - y_i)^2.$$
(19)

*Proof of Lemma 1.* To simply the notation, we drop the T subscript throughout the proof of this lemma. Let  $d_{\phi}(f,g)$  be the functional Bregman divergence [3]:

$$d_{\phi}(f,g) = \phi(f) - \phi(g) - \nabla \phi(g;f-g), \qquad (20)$$

where  $\nabla \phi(g; .)$  is the Fréchet derivative of  $\phi$  at g. Since f and  $f^{i}$  are minimizers of R and  $R^{i}$  respectively, we have:  $\nabla R(f; .) = 0$  and  $\nabla R^{i}(f^{i}; .) = 0$ . We thus have:

$$d_R(f^{\setminus i}, f) + d_{R^{\setminus i}}(f, f^{\setminus i}) = R(f^{\setminus i}) - R(f) + R^{\setminus i}(f) - R^{\setminus i}(f^{\setminus i})$$
(21)

$$= \frac{1}{m}\ell_i(f^{\setminus i}) - \frac{1}{m}\ell_i(f), \qquad (22)$$

where the last line follows by the definition of R and  $R^{i}$ . By non-negativity and additivity of divergence  $(d_{A+B} = d_A + d_B)$  we have:

$$0 \leq d_{\hat{R}^{\setminus i}}(f, f^{\setminus i}) + d_{\hat{R}^{\setminus i}}(f^{\setminus i}, f)$$

$$(23)$$

$$= -\lambda d_{\|.\|_{k}^{2}}(f, f^{\setminus i}) - \lambda d_{\|.\|_{k}^{2}}(f^{\setminus i}, f) + d_{R^{\setminus i}}(f, f^{\setminus i}) + d_{R^{\setminus i}}(f^{\setminus i}, f)$$

$$(24)$$

$$= -\lambda d_{\|.\|_{k}^{2}}(f, f^{\setminus i}) - \lambda d_{\|.\|_{k}^{2}}(f^{\setminus i}, f) + d_{R^{\setminus i}}(f, f^{\setminus i}) + d_{R}(f^{\setminus i}, f) - \frac{1}{m} d_{\ell_{i}}(f^{\setminus i}, f)$$
(25)

$$= -\lambda d_{\|\cdot\|_{k}^{2}}(f, f^{\setminus i}) - \lambda d_{\|\cdot\|_{k}^{2}}(f^{\setminus i}, f) + \frac{1}{m}\ell_{i}(f^{\setminus i}) - \frac{1}{m}\ell_{i}(f) - \frac{1}{m}d_{\ell_{i}}(f^{\setminus i}, f)$$
(26)

$$= -\lambda d_{\|.\|_{k}^{2}}(f, f^{\setminus i}) - \lambda d_{\|.\|_{k}^{2}}(f^{\setminus i}, f) + \frac{1}{m} \nabla \ell_{i}(f; f^{\setminus i} - f),$$
(27)

where line (26) is by the derivation in line (22), and line (27) is by the definition of the Bregman divergence. In the RKHS space, we have  $d_{\|.\|_k^2}(g,g') = \|g - g'\|_k^2$ , and by assumption of Theorem 3 we have  $\forall x : |g(x)| \le \kappa \|g\|_k$ . Substituting the Fréchet derivative in the above inequality, we get:

$$\|f - f^{\setminus i}\|_{k}^{2} \leq \frac{1}{\lambda m} (f^{\setminus i}(x) - f(x))(f(x_{i}) - y_{i})$$
(28)

$$\leq \frac{\kappa}{\lambda m} \|f^{\setminus i} - f\|_k (f(x_i) - y_i).$$
<sup>(29)</sup>

Cancelling the sides and squaring both sides, we get for all *x*:

$$(f(x) - f^{i}(x))^2 \leq \kappa^2 ||f - f^{i}||_k^2$$
  
(30)

$$\leq \frac{\kappa^2}{\lambda^2 m^2} (f(x_i) - y_i)^2. \tag{31}$$

Proof of Theorem 3. Let  $V = \ell_{\gamma}(f_T(X), Y) - \ell_{\gamma}(f_{T'}(X), Y)$ . Define  $V_i, 1 \le i \le 2m$  as:

$$V_{i} = \begin{cases} \ell_{\gamma}(f_{T}^{\setminus i}(X), Y) - \ell_{\gamma}(f_{T'}(X), Y) & \text{if } i \leq m \\ \ell_{\gamma}(f_{T}(X), Y) - \ell_{\gamma}(f_{T'}^{\setminus (i-m)}(X), Y) & \text{if } i > m \end{cases}$$
(32)

It is easy to see that  $\mathbb{E}_{T,T'\sim\mathcal{D}^m}[V] = 0$ . Using the concentration inequality of Theorem 6 from [4] on V and  $V_i$ , the symmetry of the training algorithm, and the symmetry of V on T and T' we get:

$$\mathbb{E}_{\substack{T,T'\sim\mathcal{D}^m\\(X,)\sim\mathcal{D}}} [(\ell_\gamma(f_T(X),Y) - \ell_\gamma(f_{T'}(X),Y))^2] = \mathbb{Var}_{\substack{T,T'\sim\mathcal{D}^m\\(X,Y)\sim\mathcal{D}}} [V] \tag{33}$$

$$\leq \sum_{i=1}^{2m} \mathop{\mathbb{E}}_{\substack{T,T'\sim\mathcal{D}^m\\(X,Y)\sim\mathcal{D}}} [(V-V_i)^2]$$
(34)

$$= 2 \mathop{\mathbb{E}}_{\substack{T,T'\sim\mathcal{D}^m\\(X,Y)\sim\mathcal{D}}} \left[ \sum_{i=1}^m (V-V_i)^2 \right]$$
(35)

$$= \frac{2}{\gamma^2} \mathop{\mathbb{E}}_{\substack{T,T'\sim\mathcal{D}^m\\(X,Y)\sim\mathcal{D}}} \left[ \sum_{i=1}^m (f_T(X) - f_T^{\setminus i}(X))^2 \right], (36)$$

where line (36) is by Lipschitz continuity of  $\ell_{\gamma}$ . Applying Lemma 1 to RHS completes the proof.  $\Box$ 

# **D** Further Experimental Details

Table 1 includes further details on the datasets used for experiments presented in the paper.

	Nomao [5]	News Popularity [6]	Twitter Buzz [7]	
# Features	89 continuous 31 nominal some missing values	61 features no missing values	77 features evolution of 11 primatry features through time no missing values	
# Samples	34,465	39,797	Sub-sampled 46,902	
Goal	predict if two business entities are the same	two business predict if a news will be predict if a tweet is go- shared more than 1400 ing to be popular times		
$\overline{T_A}$	4000 samples drop first 5 features	8000 samples drop the 3 features: self_reference_min self_reference_max self_reference_avg	4000 samples drop last 7 features	
$\overline{T_B}$	5000 samples all the features	10000 samples all the features	5000 samples all the features	
Validation Set	1000 samples	1000 samples	1000 samples	
Testing Set	28465 samples	28797 samples	45402 samples	

Table 1: Full details of the datasets used in the experimental analysis.

We optimized the hyper-parameters of each algorithm for each datasets on the validation set. Details of the chosen hyper-parameters for each algorithm is included in Table 2. The names of the parameters match the names used in Scikit-Learn [8].

Table 2: We summarize here the regularization parameters used to train the models. These parameters have been selected using a validation set of 1000 samples.

	$\underset{\alpha}{\mathbf{Ridge}}$	RFT-Regression min_weight_fraction_leaf	SVM C	Adaboost learning_rate	LinearSVR C
Nomao	0.02	0.0001	10	1.5	0.5
News	2	0.01	1.5	5	10
Twitter-Buzz	1	0.002	50	1.0	75

Full results for all experiments are included in Table 3. We have included further results on linear SVM and AdaBoost (boosted stumps). However, note that there is a regression in accuracy between the two versions of the model for the baseline algorithm. We believe that that our hyper-parameter optimization did not find a good solution for these algorithms (likely resulting in over-fitting), or that we could not effectively use the implementation in Scikit-Learn [8].

Table 3: Experiment results on 3 domains with 5 different training algorithms for a single step RCP and the MCMC methods. For the MCMC experiment, we report the numbers with the standard deviation over the 40 runs of the chain.

		Baseline	RCP	MCMC, $k = 30$	MCMC, $k = 30$	
			No RCP, No Chain	$\alpha = 0.5, \epsilon = 0.5$	$\alpha = 0.5, \epsilon = 0.5$	$\alpha = 0.7, \epsilon = 0.1$
		WLR	1.24	1.40	1.31	1.60
	Ridge	$p_{win}$	26.5	49.2	36.5	73.9
	Ruge	$C_r$	1.00	0.54	$0.54\pm0.06$	$0.32\pm0.05$
		Acc $V_1 / V_2$	93.1 / 93.4	93.1 / 93.4	$93.2 \pm 0.1$ / $93.4 \pm 0.1$	$93.0 \pm 0.3 \ / \ 93.2 \pm 0.2$
		WLR	1.02	1.13	1.09	1.12
	RF	$p_{win}$	5.6	13.4	9.8	13.1
		$C_r$	1.00	0.83	$0.83\pm0.05$	$0.59 \pm 0.05$
		Acc $V_1 / V_2$	94.8 / 94.8	94.8 / 95.0	$94.9 \pm 0.2$ / $95.0 \pm 0.2$	$94.7 \pm 0.2$ / $94.8 \pm 0.2$
0		WLR	0.79	0.79	0.00	0.00
ma	AdaBoost	p <sub>win</sub>	0.2	0.2	0.0	0.0
ĭ		$C_r$	1.00	1.00	$0.01 \pm 0.00$	$0.00 \pm 0.00$
		Acc $v_1 / v_2$	83.// 83.3	85.//85.5	$75.7 \pm 2.4$ 7 7 $3.0 \pm 2.2$	$7.4 \pm 0.2$ 7 7 7 $7.4 \pm 0.2$
		WLK	0.04	0.89	0.90	2.00
	LinSVM	$p_{win}$	1.00	1.2	1.3 0.76 $\pm$ 0.02	0.22 ± 0.02
		$\Delta c c V_r / V_r$	00.1 / 86.2	0.75	$0.70 \pm 0.02$ 001 $\pm$ 0.4 / 80 $4 \pm$ 0.3	$0.22 \pm 0.02$
		WLR	1 70	2 51	$90.1 \pm 0.4 7 \ 67.4 \pm 0.3 \\ 2 \ 32$	$\frac{90.1 \pm 0.3}{2.08}$
			82.5	99 7	99.2	97.1
	SVM	$P_{win} = C_r$	1 00	0.75	$0.69 \pm 0.06$	$0.54 \pm 0.03$
		Acc $V_1 / V_2$	94.6 / 95.1	94.6 / 95.2	$94.8 \pm 0.2$ / $95.3 \pm 0.1$	$94.9 + 0.2 / 95.2 \pm 0.1$
		WLR	0.95	0.94	1 04	0.97
		n <sub>win</sub>	2.5	2.4	6,7	3.4
	Ridge	$C_r$	1.00	0.75	$0.78 \pm 0.04$	$0.42 \pm 0.06$
		Acc $V_1$ / $V_2$	65.1 / 65.0	65.1 / 65.0	$65.0 \pm 0.1$ / $65.1 \pm 0.1$	$64.7 \pm 0.2$ / $64.7 \pm 0.2$
		WLR	1.07	1.02	1.10	1.24
	DE	$p_{win}$	8.5	5.7	10.8	26.6
	Kr	$C_r$	1.00	0.69	$0.67\pm0.04$	$0.04 \pm 0.04$
		Acc $V_1$ / $V_2$	64.5 / 65.1	64.5 / 64.7	$64.3\pm0.3$ / $64.8\pm0.2$	$63.0 \pm 0.4 \ / \ 63.0 \pm 0.4$
		WLR	0.72	0.72	0.81	0.00
SWS	AdaBoost	$p_{win}$	0.0	0.0	0.3	0.0
Ň	Tuaboost	$C_r$	1.00	1.00	$7.88 \pm 12.07$	$0.03 \pm 0.06$
		Acc $V_1 / V_2$	59.3 / 59.2	59.3 / 59.2	$59.4 \pm 0.2 \ / \ 59.2 \pm 0.0$	58.7 ± 1.1 / 58.7 ± 1.1
		WLR	0.81	1.24	1.03	1.02
	LinSVM	$p_{win}$	0.5	26.4	6.1	5.5
		$U_r$	1.00	0.90	$1.10 \pm 0.19$	$1.12 \pm 0.20$
		Acc $V_1 / V_2$	03.3 / 02.3	03.3 / 04.1	$63.5 \pm 0.5 / 05.0 \pm 0.5$	$\frac{63.0 \pm 0.8 / 03.1 \pm 0.7}{1.25}$
		WLK	1.17	1.20	1.24	1.23
	SVM	$P_{\text{win}}$	1 00	0.77	$0.86 \pm 0.02$	$0.61 \pm 0.02$
		$\Delta cc V_1 / V_2$	64 9 / 65 4	649/654	$64.8 \pm 0.1 / 65.4 \pm 0.1$	$64.7 \pm 0.2$ / $65.1 \pm 0.1$
		WID	1 71	2 54	1 52	1 50
		WLK	1./1 83.1	100.0	1.35	71.0
	Ridge	$C_{\rm win}$	1.00	0.85	$0.65 \pm 0.05$	$0.44 \pm 0.04$
		Acc $V_1 / V_2$	897/899	897/900	$901 \pm 0.03 \pm 0.03$	897 + 01 / 897 + 01
		WLR	1.35	1.15	1.15	1.03
		Dwin	41.5	16.1	15.9	6.0
	RF	$C_r$	1.00	0.86	$0.77\pm0.07$	$0.42\pm0.10$
		Acc $V_1 / V_2$	96.2 / 96.4	96.2 / 96.3	96.3 ± 0.1 / 96.3 ± 0.1	$96.2 \pm 0.1$ / $96.2 \pm 0.1$
zzr		WLR	0.93	0.90	1.13	1.17
Ē	AdaBoost	$p_{win}$	1.8	1.2	13.3	18.4
itte	Adaboost	$C_r$	1.00	1.03	$0.80\pm0.18$	$0.22\pm0.07$
M		Acc $V_1 / V_2$	95.0 / 95.0	95.0 / 95.0	$94.2 \pm 0.4$ / $94.2 \pm 0.4$	$95.5 \pm 0.3$ / $95.5 \pm 0.3$
	LinSVM	WLR	0.22	2.66	3.71	3.82
		$p_{win}$	0.0	99.9	100.0	100.0
		$C_r$	1.00	0.52	$0.61 \pm 0.45$	$0.41 \pm 0.22$
		Acc $V_1 / V_2$	94.8 / 91.2	94.8 / 96.2	$92.2 \pm 2.7 / 92.7 \pm 2.5$	$93.0 \pm 2.0 / 93.2 \pm 2.0$
		WLR	1.35	1.77	1.55	1.33
	SVM	$p_{win}$	42.2	80.0	0 70 - 0 02	<u> </u>
		$\Delta_{r}$	1.00	06.0 / 06.1	$0.70 \pm 0.03$	$0.30 \pm 0.03$
		Acc $v_1 / v_2$	90.0 / 90.1	90.0 / 90.1	$90.1 \pm 0.1$ / $90.2 \pm 0.1$	$90.1 \pm 0.1$ / $90.2 \pm 0.1$

### E Link between the accuracies, the WLR, and the Churn

Given two classifiers  $f_A$  and  $f_B$  (that is, any measurable function from  $\mathbb{R}^d$  to  $\{-1, 1\}$ ), we define the Win/Loss Ratio (*WLR*) to be  $\frac{p}{1-p}$  with  $p = \Pr[f_B(X) = Y \mid f_B(X) \neq f_A(X)]$ : p is the probability that model  $f_B$  is correct knowing that models  $f_A$  and  $f_B$  are giving a different answer.

Recall that the Churn between  $f_A$  and  $f_B$  is defined to be:

$$C = \Pr[f_A(X) \neq f_B(X)],$$

and that the accuracies of  $f_A$  and  $f_B$  are given by:

$$Acc_A = \Pr[f_A(X) = Y], \ Acc_B = \Pr[f_B(X) = Y].$$

**Lemma 2.** The relation between the WLR, Churn and accuracies of the classifiers  $f_A$  and  $f_B$  is the following:

$$WLR = \frac{p}{1-p}, \ p = \frac{1}{2} + \frac{Acc_B - Acc_A}{2C}.$$
 (37)

Proof.

$$Acc_B = \Pr[f_B(X) = Y, f_B(X) \neq f_A(X)] + \Pr[f_B(X) = Y, f_B(X) = f_A(X)].$$

Futhermore:

$$\Pr[f_B(X) = Y, f_B(X) = f_A(X)] = \Pr[f_A(X) = Y, f_B(X) = f_A(X)] = Acc_A - \Pr[f_A(X) = Y, f_A(X) \neq f_B(X)].$$

Thus using the Bayes' theorem we deduce that:

$$Acc_B = pC + Acc_A - (1-p)C,$$

which gives the result.

As discussed before, (37) confirms that for a fixed accuracy gain, less Churn is better as it increases p, and thus increases the WLR.

A statistical hypothesis test is usually used to decide if model  $f_B$  is statistically significantly better than  $f_A$ . In this setting, increasing the ratio  $\frac{Acc_B - Acc_A}{C}$  increases uniformly the power of such test.

### References

- [1] O. Bousquet and A. Elisseeff. Stability and generalization. *Journal of Machine Learning Research*, 2(Mar):499–526, 2002.
- [2] O. Bousquet and A. Elisseeff. Algorithmic stability and generalization performance. In Advances in Neural Information Processing Systems 13: Proceedings of the 2000 Conference, volume 13, page 196. MIT Press, 2001.
- [3] B. A. Frigyik, S. Srivastava, and M. R. Gupta. Functional Bregman divergence and Bayesian estimation of distributions. *Information Theory, IEEE Transactions on*, 54(11):5130–5139, 2008.
- [4] S. Boucheron, G. Lugosi, and O. Bousquet. Concentration inequalities. In Advanced Lectures on Machine Learning, pages 208–240. Springer, 2004.
- [5] L. Candillier and V. Lemaire. Design and analysis of the Nomao challenge active learning in the real-world. In Proceedings of the ALRA: Active Learning in Real-world Applications, Workshop ECML-PKDD, 2012.
- [6] K. Fernandes, P. Vinagre, and P. Cortez. Progress in Artificial Intelligence: 17th Portuguese Conference on Artificial Intelligence, EPIA 2015, Coimbra, Portugal, September 8-11, 2015. Proceedings, chapter A Proactive Intelligent Decision Support System for Predicting the Popularity of Online News, pages 535–546. Springer International Publishing, Cham, 2015.
- [7] F. Kawala, E. Gaussier, A. Douzal-Chouakria, and E. Diemert. Apprentissage d'ordonnancement et influence de l'ambiguïté pour la prédiction d'activité sur les réseaux sociaux. In *Coria*'2014, pages 1–15, Nancy, France, France, March 2014.
- [8] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay. Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, 12:2825–2830, 2011.