Шугаман SVM ашиглан хорт програм илрүүлэх асуудалд

С. Байгалтөгс, Ч. Эрдэнэбат
ШУТИС, КТМС, Компьютерийн ухааны салбар
baigaltugs@must.edu.mn

***Хураангуй*—Өнөөдөр дэлхийн хөгжингүй улс орнууд мэдээллийн технологийн салбартаа дор бүрнээ, өөрийн, үндэсний анти-вирус програмуудыг бодлогоор зориуд хөгжүүлж .. хэрэглэж иржээ. Хөгжиж буй орнууд ч энэ чиг хандлагыг хүлээн зөвшөөрч "даган баясч" буй нь зүйн хэрэг юм. Тэд өөр өөрсдийн анти-вирус програмуудыг идэвхийлэн хөгжүүлсээр байна. Манай орны хувьд мэдээллийн салбарын энэхүү "стратегийн технологи" –ийг бусдын нэгэн адил хир чинээгээр хөгжүүлж байгаа ба ойрын ирээдүйд өөрийн гэсэн анти-вирус програмын сантай болох нь улс орны үндэсний аюулгүй байдлын зорилтуудын нэг болж байна. Хорт програмаас хамгаалахын тулд эхлээд тэдгээрийг илрүүлэх учиртай. Бид өгүүлэлдээ дата майнингийн аргыг ашиглан хорт програмуудыг илрүүлэх асуудлыг тухайлан авч үзэв. Өгүүлэлд, шугаман SVM алгоритм ашиглан хорт програм илрүүлэх туршилт .. түүний үр дүнг тоймлосон болно. Туршилтаар, шугаман SVM алгоритм нь хангалттай тооны хорт програмын цуглуулга байж гэмээнэ .. тэдгээрээс суралцаад шинэ буюу нэр үл мэдэгдэх хорт програмуудыг илрүүлэлтийг 74 – 83% -ийн магадлалтайгаар хэрэгжүүлэх боломжтой гэдгийг нотолсон гэж үзнэ.**

Түлхүүр үг—malware detection; data mining; svm; support vector machine; linear svm;

I. Удиртгал

Компьютерийн технологийн хөгжил, түүний хэрэглээний өсөлтийг дагаад хорт програмын тархац нэмэгддэг зүй тогтолтой. Компьютерт зөвшөөрөлгүй нэвтрэх чадвартай, "хэрэг тарих" тодорхой чиг үүрэг бүхий код – бичвэрийг бүхэлд нь "хорт програм" ухагдахуунд хамаатуулж болно [1]. Хорт програм буюу malware гэдэг нь malicious software гэсэн үгний товчлол юм [2]. Хорт програмын өсөлт хөгжил нь өнөө үед илрүүлэлтийн технологиудыг эрс сайжруулахыг шаардаж байдаг. 2003 оны сүүлээр гарсан, дэлхийн мэдээлэл, холбооны технологийн хөгжлийг тодорхойлож байгаа 30 гол тэргүүлэх чиглэлийн тоонд вирусээс хамгаалах технологи багтсан [3], [4].

Хорт програм илрүүлэх олон арга бий. Өнөөдөр, анти-вирус програмууд их төлөв хорт програм илрүүлэлтийн "сигнатурт тулгуурласан аргачлал" дээр оршин тогтнож байна [5]. Энэ нь шинэ буюу үл мэдэгдэх вирусыг тухайн цагт нь шуурхай соргог илрүүлж үл чадах байдлыг нөхцөлдүүлдэг. Сигнатурт тулгуурлаагүй илрүүлэлтийн аргуудын нэг нь дата майнинг (англиар) буюу өгөгдлийн тандалтын арга юм. Дата майнингийн арга нь машины сургалтанд ашиглагддаг. Өөрөөр хэлбэл их хэмжээний өгөгдлөөс зүй тогтол илрүүлж, бүртгэх, шинжлэх замаар [холбогдох шинэ] мэдлэгийг бий болгон бүтээж, түүгээрээ машиныг өөрийг нь бүтээлчээр ажиллуулж сургах явдал юм. Суралцах процесс нь ангилал болгоны хувьд тухайн ангилалд хамаатах бичлэгүүд, тэдгээрийн шинж чанарыг илэрхийлэх аттрибутууд (feature) дээр явагдана [6].

Энэхүү өгүүлэлд "машин сургалтын шугаман SVM алгоритм" –ыг ашиглан хорт програм илрүүлэх аргачлалын талаар авч үзлээ. SVM алгоритм нь машин сургалтын шилдэг алгоритмуудын нэг бөгөөд кернель гэгдэх цөмөөсөө хамаарч ялгамжтай үр дүнг үзүүлнэ. Жирийн SVM нь их хэмжээний өгөгдөл дээр төдийлэн тохиромжтой биш байдаг бол харин, шугаман SVM нь их хэмжээний өгөгдөл дээр ажиллах бололцоотойгоороо давуулаг .. онцлог юм.

Шугаман SVM алгоритмын талаар бид 2 зүйлд анхаарсан. Өгөгдлийн олонлог үүсгэх, мөн, өгөгдлийн шугаман алгоритмын талаар, тэдгээрт холбогдох туршилт, үр дүнгийн талаар доорхи дэд гарчигуудын хүрээнд авч үзэв. Товчдоо, өгүүллээр бид, хорт програм илрүүлэхэд шугаман SVM алгоритмыг ашиглан үр дүнд хүрэх бололцоотой гэдгийг харуулав.

II. Шугаман SVM алгоритм

SVM (Support Vector Machine) нь өгөгдөл ангилахад хэрэглэгддэг техник юм. Өгөгдөл ангилах ажиллагаа нь ихэвчлэн сургалтын ба туршилтын гэсэн өгөгдлийн олонлогуудыг хамааруулна. Өгөгдлийн олонлогын элемент болгон хэд хэдэн аттрибут (feature) ба ангилалын утгаас тогтдог. SVM –ын зорилго бол туршилтын өгөгдлийн олонлогын тухайн нэг элементийн .. дөнгөж [зөвхөн] аттрибутууд нь өгөгдсөн байхад [тохиолдолд] түүний ангилалыг таамаглах загвар үүсгэх явдал юм. Уг алгоритм нь олон янзын кернельтэй байж болох ч, тэдгээрт, үндсэндээ шугаман, полиноминаль, RBF, сигмойд гэсэн дөрвөн төрөл зүйл гололдог [7].

Шугаман SVM нь олон тооны аттрибут бүхий их хэмжээний өгөгдөл дээр тун сайн ажилладаг алгоритм юм [8]. Практикт 10000 –аас дээш хэмжээний өгөгдөл дээр шугаман бус SVM удаашралтай нь нэгэнт тогтоогджээ. Иймд, судалгаандаа бид шугаман SVM –ийг ашиглахаар шийдсэн.

III. Өгөгдлийн олонлог үүсгэх нь

Бид эхний ээлжинд VX Heaven [9] сайтын эмхтгэсэн 271094 хорт програмын цуглуулгыг татаж авав. Уг хорт програмуудаас шахагдаагүй [10] 51243 хорт програм болон гэм хоргүй 1560 програм нэгтгэн нийлүүлж 52803 элемент бүхий өгөгдлийн олонлог үүсгэлээ. Энэхүү өгөгдлийн олонлогт буй програмуудаас бид текстүүдийг нь ялган авав. Ялгаж авсан текстүүдийг зохих ёсоор векторжуулсан. Ерөнхийдөө, програм x бүр |F| урттай вектороор дүрслэгдэнэ:

$$<w1(x), w2(x), w3(x), ..., wF(x)>$$

Энд wj(x) нь j –дэх үгийн жин юм.

Жигнэх давтамж (frequency), TFIDF зэрэг янз бүрийн арга байдаг [11]. Судалгаандаа давтамж болон TFIDF аргуудыг хослуулж ашигласан. Давтамж ашиглан вектор үүсгэх ийм аргачлалыг bag-of-words хэмээнэ [12]. Давтамж тооцоолох ажиллагаанд үг бүрийг тоолохоос гадна биграм гэж нэрлэгддэг дараалсан үгсийн хослолуудыг мөн тоолж векторжуулсан. Манай үүсгэсэн өгөгдлийн олонлог нь нийтдээ 40 төрлийн [доорхи] ангилалтай.

Хүснэгт 1. Өгөгдлийн олонлогын ангилал, тэдгээрт оногдох програмын тоо

№	Ангилал	Тоо	№	Ангилал	Тоо
1	trojan-psw	2438	21	hacktool	232
2	trojan-clicker	686	22	trojan-mailfinder	45
3	worm	967	23	virtool	165
4	hoax	248	24	virus	2306
5	not-virus:hoax	19	25	trojan-spy	1872
6	exploit	247	26	spoofer	7
7	rootkit	1290	27	flooder	126
8	trojan-downloader	11545	28	email-flooder	102
9	trojan-im	146	29	irc-worm	93
10	trojan-ransom	9	30	trojan	10375
11	packed	57	31	trojan-ddos	61
12	trojan-proxy	276	32	trojan-dropper	2286
13	trojan-gamethief	1539	33	email-worm	923
14	benign	1560	34	spamtool	15
15	im-flooder	72	35	net-worm	183
16	trojan-banker	1088	36	p2p-worm	198
17	trojan-notifier	26	37	constructor	277
18	backdoor	11058	38	not-virus:badjoke	12
19	sms-flooder	31	39	dos	113
20	im-worm	109	40	sniffer	1

IV. Туршилт ба дүгнэлт

Туршилтын явцад, нийт өгөгдлийн олонлогоо сургалтын ба тестийн гэсэн хоёр хэсэгт хуваав. Хуваахдаа 67 хувийг нь сургалтанд, үлдсэн 33 хувийг нь тестэнд ашиглалаа. Туршилтыг нийт 4 удаа хийв. Туршилт болгондоо үндсэн өгөгдлийн олонлогоос санамсаргүй сонголтоор сургалтын болон тестийн хэсгүүдээ гаргаж авч байв. Сургалтын өгөгдлийн олонлогоо ийн гаргаж авсны дараагаар шугаман SVM алгоритмыг ашиглан машиныг зорилтот үйлдэлд сургасан. Үлдэх 33% болох тестийн өгөгдлөө [нэгэнт] сургасан машинаараа таамаглуулсан. Туршилт явуулахдаа бид, програм бүрд 297003 аттрибут бүхий вектор үүсгэсэн болно. (Дөрвөн туршилтын өгсөн дундаж дүнг дараахь хүснэгтээс харна уу. Хүснэгт - 2)

Хүснэгт 2. Туршилтын үр дүн

№	Ангилал	4 туршилтын дундаж үр дүн (аттрибут: 297003)			
		precision	**recall**	**f1-score**	**support**
1	trojan-psw	0.68	0.65	0.66	788
2	trojan-clicker	0.72	0.62	0.67	213
3	worm	0.61	0.53	0.57	329
4	hoax	0.57	0.39	0.46	74
5	not-virus:hoax	0	0	0	2
6	exploit	0.43	0.43	0.43	69
7	rootkit	0.79	0.84	0.82	419
8	trojan-downloader	**0.83**	**0.81**	**0.82**	3801
9	trojan-im	0.61	0.57	0.59	47
10	trojan-ransom	0.05	0.2	0.08	5
11	packed	0.17	0.04	0.06	25
12	trojan-proxy	0.56	0.46	0.5	85
13	trojan-gamethief	0.58	0.89	0.7	522
14	benign	0.91	0.87	0.89	507
15	im-flooder	0.47	0.32	0.38	22
16	trojan-banker	0.75	0.82	0.78	355
17	trojan-notifier	0.25	0.17	0.2	6
18	backdoor	**0.81**	**0.85**	**0.83**	3706
19	sms-flooder	0.29	0.25	0.27	8
20	im-worm	0.68	0.57	0.62	30
21	hacktool	0.37	0.32	0.34	71
22	trojan-mailfinder	0.29	0.31	0.3	16
23	virtool	0.34	0.17	0.22	60
24	virus	0.77	0.75	0.76	750
25	trojan-spy	0.66	0.58	0.62	628
26	spoofer	0	0	0	3
27	flooder	0.39	0.29	0.33	38
28	email-flooder	0.54	0.5	0.52	30

29	irc-worm	0.7	0.48	0.57	33
30	trojan	**0.74**	**0.75**	**0.74**	3433
31	trojan-ddos	0.2	0.05	0.08	21
32	trojan-dropper	0.58	0.53	0.55	729
33	email-worm	0.72	0.67	0.69	326
34	spamtool	0.6	0.43	0.5	7
35	net-worm	0.56	0.4	0.47	60
36	p2p-worm	0.67	0.66	0.66	58
37	constructor	0.62	0.5	0.55	101
38	not-virus:badjoke	0	0	0	5
39	dos	0.56	0.33	0.42	42
40	sniffer	0	0	0	1
	Average / Total	**0.75**	**0.75**	**0.75**	**17425**

Эдгээр туршилтаас үзвэл хорт програм илрүүлэхэд шугаман SVM алгоритмыг үр дүнтэй бүтээлчээр ашиглаж болох нь тод харагдаж байна. Танилтын дундаж хувь ~75% байгаа нь чамлахааргүй үзүүлэлт мөн гэж үзнэ. Зарим ангилал дээр бидний туршилт бага хувьтай гарсан нь тухайн ангилалд хамаатах сургалтын өгөгдөл хэт бага байсантай холбоотой болов уу. Ерөнхийдөө, машин сургах өгөгдлийн тоо, тухайн ангилал нэгбүр дээр дор хаяж 10000 –аас дээш байх нөхцөлд шугаман SVM алгоритм нь [сая, өмнө дурдсан] 74 – 83% -ийн магадлалтай танилт өгөх боломжтойг манай судалгаа харуулж байна. Мөн энэ удаагийн туршилт 297003 аттрибутуудтайгаар явагдсан. Хэрэв бид аттрибутууд дээр үнэлгээ хийж, жин багатайнуудыг нь ялгаж хасвал зөв танилтын хувь дээшлэн нэмэгдэж өсөх боломжтой гэж үзэж байна. Цаашдын судалгаандаа бид аттрибут сонголт дээр нарийвчлал хийж, танилтын хувь процентийг өсгөн нэмэгдүүлэх чиглэлд ажиллах болно.

Монголын холбогдох салбарын удирдлагын шийдвэр гаргах дээд түвшинд мэдээллийн аюулгүй байдалд түлхүү анхаарах, дотоодын үйлдлийн систем .. дотоодын анти-вирус шийдлүүдийг хөхүүлэн дэмжихээ нэгэнт илэрхийлсэн [13]. Өмнө нь хорт програм илрүүлэх чиглэлээр, монголд, судалгааны ажил тэр бүрий хийгдэж байгаагүй юм. Энэ судалгааны ажил нь манайд өөрийн анти-вирус програмыг хөгжүүлж тулгамдсан хэрэгцээгээ хангах боломж бололцоо бүхий болохыг харуулж байгаагаараа онцлог юм.

НОМ ЗҮЙ

[1] Ч. Эрдэнэбат, А. Ундармаа, “Мулти-паттерн хайлт ашиглан хорт програм илрүүлэх арга,” in *Холбоо, мэдээллийн технологийн өнөө ба ирээдүй - 2013*, 2013, pp. 145–154.

[2] “Reversing Malware,” *Drexel University*, 2008. [Online]. Available: https://www.cs.drexel.edu/~spiros/teaching/CS675/slides/malware.pdf. [Accessed: 29-Nov-2012].

[3] B. Sukhbaatar, “Introduction of new Information and Communication Technologies in Mongolia,” in *International Forum on Strategic Technology (IFOST 2007)*, 2007.

[4] Б. Сүхбаатар, Л. Батхишиг, “Цахилгаан холбооны салбарт гарч буй шинэ технологиуд, түүнийг Монгол улсад судлан нэвтрүүлэх зарим асуудлууд,” *Билэг*, Улаанбаатар, p. 8, Jan-2007.

[5] M. Feng and R. Gupta, “Detecting virus mutations via dynamic matching,” in *IEEE International Conference on Software Maintenance 2009*, 2009, pp. 105–114.

[6] J. Han, M. Kamber, and J. Pei, *Data Mining: Concepts and Techniques, Third Edition (The Morgan Kaufmann Series in Data Management Systems)*. Morgan Kaufmann, 2011, p. 744.

[7] C.-W. Hsu, C.-C. Chang, and C.-J. Lin, “A Practical Guide to Support Vector Classification,” Taipei, Apr. 2010.

[8] R.-E. Fan, K.-W. Chang, C.-J. Hsieh, X.-R. Wang, and C.-J. Lin, “LIBLINEAR: A Library for Large Linear Classification,” *Journal of Machine Learning Research*, vol. 9, pp. 1871–1874, 2008.

[9] A. Baranovich, “VX Heavens,” *Vx Heavens*, 2012. [Online]. Available: http://vx.netlux.org/index.html. [Accessed: 25-Mar-2013].

[10] Ч. Эрдэнэбат, С. Байгалтөгс, “Энтропи хэмжигдэхүүн ашиглан хорт програмыг шинжлэх зарим асуудалд,” in *“Залуу судлаач - Хөгжлийн гарц” ЭШБХ, Эрдэм шинжилгээний бүтээлийн эмхэтгэл №1/103*, 2009, pp. 74–79.

[11] P. Soucy and G. W. Mineau, “A Simple KNN Algorithm for Text Categorization,” in *IEEE International Conference on Data Mining (ICDM 2001)*, 2001, pp. 647–648.

[12] Z. Harris, “Distributional structure,” *Word*, vol. 10, pp. 146 – 162, 1954.

[13] “‘НЭЭЛТТЭЙ ЭХИЙН ПРОГРАММ ХАНГАМЖИЙГ НЭВТРҮҮЛЭХ, ХӨГЖҮҮЛЭХ БОЛОМЖ’ СЭДЭВТ ХЭЛЭЛЦҮҮЛГЭЭС ГАРГАСАН ЗӨВЛӨМЖ,” *Монгол Улсын Их Хурал*, 21-Jan-2013. [Online]. Available: http://www.parliament.mn/secretariat/opensource/categories/2477/pages/3237. [Accessed: 23-Jan-2013].