2012 11th International Conference on Machine Learning and Applications

On the Validity of a New SMS Spam Collection

José María Gómez Hidalgo
R&D Department
Optenet
Las Rozas, Madrid, Spain
jgomez@optenet.com

Tiago A. Almeida
Department of Computer Science
Federal University of São Carlos – UFSCar
Sorocaba, São Paulo, Brazil
talmeida@ufscar.br

Akebo Yamakami
School of Electrical and Computer Engineering
University of Campinas – UNICAMP
Campinas, São Paulo, Brazil
akebo@dt.fee.unicamp.br

***Abstract*—Mobile phones are becoming the latest target of electronic junk mail. Recent reports clearly indicate that the volume of SMS spam messages are dramatically increasing year by year. Probably, one of the major concerns in academic settings was the scarcity of public SMS spam datasets, that are sorely needed for validation and comparison of different classifiers. To address this issue, we have recently proposed a new SMS Spam Collection that, to the best of our knowledge, is the largest, public and real SMS dataset available for academic studies. However, as it has been created by augmenting a previously existing database built using roughly the same sources, it is sensible to certify that there are no duplicates coming from them. So, in this paper we offer a comprehensive analysis of the new SMS Spam Collection in order to ensure that this does not happen, since it may ease the task of learning SMS spam classifiers and, hence, it could compromise the evaluation of methods. The analysis of results indicate that the procedure followed does not lead to near-duplicates and, consequently, the proposed dataset is reliable to use for evaluating and comparing the performance achieved by different classifiers.**

***Keywords*-Spam filtering; Mobile spam; Text categorization; Classification; Text analysis.**

I. Introduction

Text messaging is a communication service component of phone, web or mobile communication systems, using standardized communications protocols that allow the exchange of short text messages between fixed line or mobile phone devices. While the original term was derived from referring to messages sent using the Short Message Service (SMS), it has since been extended to include messages containing image, video, and audio.

Mobile text messages are commonly used between mobile phone users, as a substitute for voice calls in situations where voice communication is impossible or undesirable. Such way of communication is also very popular because in some places text messages are significantly cheaper than placing a phone call to another mobile phone.

Messaging still dominates mobile market non-voice revenues worldwide. According to a report recently provided by Portio Research[1], the worldwide mobile messaging market was worth USD 179.2 billion in 2010, has passed USD 200 billion in 2011, and probably will reach USD 300 billion in 2014. The same study indicates that annual worldwide SMS traffic volumes rose to over 6.9 trillion at end-2010 to break 8 trillion by end-2011.

Mobile messages can be used to interact with automated systems such as ordering products and services for mobile phones or participating in contests. Service providers and advertisers use direct text marketing to notify mobile phone users about promotions, payment due dates and other notifications that can usually be sent by post or e-mail.

The downside is that cell phones are becoming the latest target of electronic junk mail, with a growing number of marketers using text messages to target subscribers. SMS spam (sometimes also called mobile phone spam) is any unwanted or unsolicited text message received on a mobile device. Although this practice is rare in North America, it has been very common in some parts of Asia.

SMS text messaging offers a target rich environment for spammers. With the explosive growth in text messaging along with unlimited texting plans it barely costs anything for the attackers to send malicious messages. This combined with the trust users inherently have in their mobile devices makes it an environment rife for attack. In fact, a recently Cloudmark company research[2] reveals that financial fraud and spam via text messages is now growing at a rate of over 300 percent year over year.

In the same way that carriers are facing a real challenge in dealing with SMS spam, academic researchers in this field are also experiencing some difficulties. Probably, one of the major concern corresponds to the lack of large, real and public databases. Unlike the large amount of available email spam datasets [1], [2], [3], [4], [5], [6], [7], there are very few corpora with real examples of mobile phone spam, and to make matters worse, they are usually of small size.

To fill this important gap, we have recently proposed the new SMS Spam Collection [8], which is a real, public, non-encoded, and the largest SMS spam corpus as far as we know. However, it have been created by augmenting a previously existing database built using roughly the same sources. Thus, it is very important to verify if there are some duplicates coming from other databases, since added messages may contain previously existing messages in the original collection. In this way, in this paper we have performed a detailed analysis of the new SMS Spam Collection in order to ensure that this does not happen, as it may ease the task of learning SMS spam classifiers.

[1] `http://www.portioresearch.com/MMF11-15.html`

[2] `http://blog.cloudmark.com/2011/12/05/surge-in-financial-related-mobile-spam-in-q4`

978-0-7695-4913-2/12 $26.00 © 2012 IEEE
DOI 10.1109/ICMLA.2012.211

This paper is organized as follows: Section II presents the new SMS Spam Collection. A comprehensive near-duplicate analysis of the new SMS Spam Collection with the main results are presented in Section III. Finally, in Section IV, we offer conclusions and outlines for future work.

II. The SMS Spam Collection

Reliable data are essential in any scientific research. It is a common sense that the absence of representative data can seriously impact the processes of evaluation and comparison of methods and unfortunately, areas of more recent studies are generally affected by the lack of public available data.

To address the lack of SMS spam datasets, in [8], we propose a new real, public and non-encoded SMS Spam Collection[3] that is the largest one as far as we know. Moreover, we offer a comprehensive performance evaluation comparing several established machine learning methods in order to provide good baseline results for further comparison.

As pointed out in [8], to create the SMS Spam Collection we have collected data derived from different sources.

First, a set of 425 SMS spam messages was manually extracted from the Grumbletext Web site[4]. This is a UK forum in which cell phone users make public claims about SMS spam messages, most of them without reporting the very spam message received. The identification of the text of spam messages in the claims is a very hard and time-consuming task, and it involved carefully scanning hundreds of web pages.

We have also added legitimate samples by inserting 450 SMS messages collected from Caroline Tag's PhD Thesis[5].

Furthermore, we have selected 3,375 SMS ham messages randomly chosen of the NUS SMS Corpus[6].

Finally, we have incorporated the SMS Spam Corpus v.0.1 Big[7] that is composed by 1,002 SMS ham messages and 322 spam messages. More detail about this dataset can be found in [9], [10], and [11]. However, it is important to point out that the sources used for building this corpus are almost the same used to create the new SMS Spam Collection.

Despite the importance of the new collection in a scenario that requires a lot of such data, the SMS Spam Collection was created with messages of previously existing database built using roughly the same sources. Therefore, at this stage, it is very important to perform a careful analysis of the validity of the proposed dataset checking if there are duplicates coming from both databases.

III. Duplicate analysis of the SMS Spam Collection

To ensure that the way the SMS Spam Collection has built, by reusing the same message sources, does not lead to invalid SMS spam filtering results, it is needed to study the potential overlap between the sub-collections that have been used when building it. The hypothesis is that the messages added to the original SMS collection, even extracted from the same sources (the Grumbletext site, the NUS SMS Corpus), do not add duplicates to those previously existing messages, except for those previously existing in the original collection or the messages sources themselves. In this way, if there are duplicates in the final collection, the only causes can be:

- Spammers do use templates when writing their spam messages.
- Legitimate users do make use of message templates existing in their mobile phones.
- Legitimate users do re-send chain letters (*e.g.* jokes, Christmas messages, etc.).

So, if the task of SMS spam filtering is eased because of these duplicate messages, the reason for this is the actual behavior of SMS messaging by spammers and legitimate users, and not the way the collection used for testing was built.

In consequence, we have built three SMS sub-collections described below (original, added and all messages), and we have studied the most frequent duplicates in all the sub-collections. The hypothesis gets confirmed if:

1) The existing duplicates in the original sub-collection keep the same frequency statistics in the final collection, and
2) the existing duplicates in the added messages keep the same frequency statistics in the final collection as well.

In the next sections, we describe the three sub-collections used in the study, along with the approach we have used to detect message duplicates, or more properly, near-duplicates. We detail the results of the analysis, which confirm our hypothesis.

A. Text collections

In order to evaluate the potential overlap between the datasets which were used to build the proposed SMS Spam Collection, we have searched for near-duplicates within three sub-collections:

- The previously existing SMS Spam Corpus v.0.1 Big (**INIT**).
- The SMS collection that includes the additional messages from Grumbletext, the NUS SMS Corpus, and the Tag's PhD Thesis (**ADD**).
- The released SMS Spam Collection (**FINAL**).

The **INIT** dataset has a total of 1,324 text messages where 1,002 are ham and 322 are spam. The **ADD** sub-collection is composed by 3,825 legitimate messages and 425 mobile spam messages, for a total of 4,250 text messages. The percentages of ham and spam are shown in Table I.

It is worth noticing that the previously existing SMS Spam Corpus v.0.1 Big, which corresponds to the **INIT** sub-collection, poses a simpler problem to machine learning content based spam filters, as the collection is more balanced than the new SMS Spam Collection. On the other side, the new collection is much bigger, and more data often implies better learning generalization.

[3]The SMS Spam Collection is available at `http://www.dt.fee.unicamp.br/~tiago/smsspamcollection`

[4]The Grumbletext Web site is available at `http://www.grumbletext.co.uk/`

[5]The Caroline Tag's PhD Thesis is available at `http://etheses.bham.ac.uk/253/1/Tagg09P hD.pdf`

[6]The NUS SMS Corpus is available at `http://www.comp.nus.edu.sg/~rpnlpir/downloads/corpora/smsCorpus/`

[7]The SMS Spam Corpus v.0.1 Big is public available at `http://www.esp.uem.es/jmgomez/smsspamcorpus/`

Table I: How the sub-collections are composed.

	INIT		ADD	
Class	Amount	Pct	Amount	Pct
Ham	1,002	75.68	3,825	90.00
Spam	322	24.32	425	10.00
Total	1,324	100.00	4,250	100.00

In Table II we present the main statistics related to the tokens extracted from the **INIT** and **ADD** sub-collections.

Table II: Basic statistics related to the tokens extracted from the sub-collections.

	INIT	ADD
Ham	12,192	51,419
Spam	7,682	9,861
Total	19,874	61,280
Avg per Msg	15.01	14.42
Avg in Ham	12.17	13.44
Avg in Spam	23.86	23.20

Note that, for both sub-collections, mobile phone spams are in average ten tokens larger than legitimate messages. Also note that the average tokens per message is quite similar in both sub-collections.

B. Near-duplicate detection approach

For the particular needs of this study, and given the short nature of SMS messages, the "String-of-Text" method can be considered as a reasonable baseline for the purpose of detecting near-duplicated messages in our collection.

The "String-of-Text" method, implemented by the WCopyfind[8] tool, involves scanning suspect texts for approximately matching character sequences. In order to avoid little manual modifications, this approximation can involve transformations like case changing, separators variation (e.g. addressing those users including more white spaces between words), etc.

The "String-of-Text" method is a simplified version of the general N-gram matching detection method, widely used in the literature [12], [13]. An N-gram is an ordered sequence of tokens or words present in a text, in which N is the number of tokens.

For this purpose, texts are compared searching for N-grams for relatively big sizes (*e.g.* N=6), with additional parameters (length of match in number of characters, etc.). This approach is implemented in WCopyfind, but we have simplified it to N-gram matches after text normalization involving:

- Replacing all token separators by white spaces.
- Lowercasing all characters.
- Replacing digits by the character 'N' (to preserve phone numbers structure).

For instance, the 6-gram "`stop to NNNNN customer services NNNNNNNNNNN`" corresponds to a match between the next two messages within the **ADD** sub-collection:

```
Thank you, winner notified by sms. Good
Luck! No future marketing reply STOP
to 84122 customer services 08450542832
```

and

```
Your unique user ID is 1172. For removal
send STOP to 87239 customer services
08708034412
```

As it can be seen, both messages are not near-duplicates; instead, they share a common pattern in messages reported by users as SMS spam in the Grumbletext site, which is the matching 6-gram. In particular, both messages correspond to two different SMS advertising campaigns in which the users have actually not subscribed the service.

In consequence, this near-duplicate approach, especially with relatively short N-Grams, can lead to many false positives. As a result, the statistics collected during our analysis represent an upper bound of the potential near-duplicates that occur in the final collection. In our opinion, this is safer than finding a lower bound, because in this way no near-duplicates will be missing, and the conclusions of the study are sound.

In order to find matching N-grams and message near-duplicates within a given sub-collection, we have followed the next procedure:

1) All messages within the sub-collection are taken as a sorted list.
2) Each N-gram for a message is built from left to right.
3) A match or hit is registered when an N-gram present in a message i is found in a message j, with $i < j$.
4) If a hit for messages i and j is registered, no other matches between those messages are stored.
5) All N-grams occurring in two or more messages are stored, along with the number of messages in which they occur.

Thus, if a particular N-gram is present in messages i, j and k with $i < j < k$, only the hits for i and j, and for j and k are counted. It must be noted that it is possible that there is a match between messages i and j, and another match between j and k, but not between i and k because both previous matching N-grams are different (although they may have some overlap). In consequence, the way we compare SMS messages is not symmetric.

It is worth noting that it may be the case that two messages have several N-grams in common. In fact, that would be the case for full long duplicate messages. In this situation, only the first left N-gram is reported, and then other co-occurring N-grams may be missing counts for yet other messages.

C. Results and analysis

The goal of this process is to check if merging the first two sub-collections adds many near-duplicates to the

[8]See: `http://plagiarism.phys.virginia.edu`

final database, in order to assess the overlap between both collections. Within each sub-collection, we have compared each pair of messages, stored all N size matches (N-grams with N = 5, 6, and 10), and sorted the N-grams according to their frequency, examining in detail the top ten ones per N. According to the literature, N = 6 is a typical number for detecting near-duplicate paragraphs, and we have tested N = 5 because some messages were exactly this long, but there are not nearly shorter messages. Moreover, while N = 5 or N = 6 can lead to many false positives, these hits can be refined with the longer matches required with N = 10, which in turn is quite close to the actual message length average.

1) Frequency results: We show the overall N-gram occurrence statistics for N = 5, 6 and 10 in the **INIT**, **ADD** and **FINAL** sub-collections in Table III. In the third column, we list the number of unique N-grams with 2 or more occurrences for a given size in each sub-collection. As it can be expected, we can view that the numbers increase with the the number of messages in each sub-collection.

Table III: N-gram occurrence statistics for different sizes in the studied sub-collections.

N	sub	#uniq	sum	avg	std
5	**INIT**	186	573	3.08	1.56
	ADD	484	1292	2.67	2.02
	FINAL	718 (+48)	2175	3.03	2.24
6	**INIT**	140	420	3.00	1.37
	ADD	361	923	2.56	1.20
	FINAL	548 (+47)	1619	2.95	1.71
10	**INIT**	92	243	2.64	0.99
	ADD	192	489	2.55	1.33
	FINAL	354 (+70)	964	2.72	1.41

We can notice as well that, typically, the number of unique N-grams for the **FINAL** sub-collection is bigger than the sum of N-grams in the **INIT** and **ADD** sub-collections. The exact number of new N-grams that is added to the **FINAL** collection is presented in parenthesis. The difference of unique new N-grams between 5- and 6-grams is small and, as expected, there are less new 6-grams than 5-grams.

However, the number of new unique 10-grams is quite bigger than previous ones, what may be considered counter-intuitive. Moreover, and due to their length, 10-grams are much less likely to correspond to false positive near-duplicates. In consequence, we have examined those 10-grams in **FINAL** occurring exactly in a message in **INIT** and in a message in **ADD** (thus, with an exact frequency of 2). We have found that 52% of them do contain "N^+" strings, representing short and/or telephone numbers in spam messages, and in consequence, the matched messages belong to the same SMS spam campaign. It must be noted that SMS messages in the same spam campaign can use different short and/or telephone numbers. The remaining 10-grams with a frequency of 2 do correspond to:

- Other spam messages (*e.g.* "u are subscribed to the best mobile content service in").
- Chain letter messages extracted from the NUS SMS Corpus (*e.g.* "the xmas story is peace the xmas msg is love").
- Actual duplicates contributed to the NUS SMS Corpus (*e.g.* "i have been late in paying rent for the past").

Regarding the rest of figures in Table III, the fourth, fifth and sixth columns report the total and the average number of hits per N-gram, plus the standard deviation, for each N-gram size and sub-collection, respectively. Only N-grams occurring in two or more messages are reported, because the N-grams considered are those that can correspond to near-duplicates. For instance, there are 573 hits of the 186 unique 5-grams with frequency of two or more messages for the **INIT** sub-collection, and each 5-gram occurs on an average of 3.08 ± 1.56 messages.

As it can be expected, the longer the N-grams, the less total number and average of matching messages, because the probability of getting a longer match between two randomly chosen messages is smaller. In general, the figures for **INIT** messages are bigger than for **ADD**, what makes sense because the proportion of spam in the first collection is three times the proportion in the second collection, and most of the N-gram matches correspond to SMS spam messages. This explains as well that the average number of matches in the **FINAL** sub-collection is closer to the **INIT** average than to the **ADD** average, as the total counts of spam messages is 322 and 425 for these latter sub-collections, respectively. As previously discussed, most matches come from spam messages, that make for the near-duplicates because of the intrinsic similarity between spam campaigns patterns, and **ADD** spam messages sum up on previously existing campaigns and patterns in the **INIT** sub-collection. In other words, the spam class messages are typically more similar among them, than the ham class, for any of the sub-collections.

2) Top scoring N-grams: In order to compare the actual matches between messages in the studied sub-collection, we report the top frequent N-grams and their frequencies for each N in the next tables. We show the ten top frequent 5 and 6-grams in Tables IV and V, respectively.

First of all, it must be noted that, given an N-gram with counts i, j and k in the **INIT**, **ADD** and **FINAL** collections respectively, we must not expect that $i + j = k$. This is because some counts are missing as a previous N-gram match between two messages may have been reported, and only N-gram matches corresponding to the left most match between two messages are summed up.

As it can be seen regarding 5-grams:

- 5-grams already present in the **INIT** and the **ADD** sub-collections do not collapse to greatly increase their frequency. For instance, the 5-grams "sorry i ll call later" and "i cant pick the phone" do not change

Table IV: Ten top 5-grams and their frequencies in the studied sub-collections.

INIT		ADD		FINAL	
5-gram	#f	5-gram	#f	5-gram	#f
we are trying to contact	10	sorry i ll call later	37	sorry i ll call later	37
this is the Nnd attempt	9	private your NNNN account statement	15	private your NNNN account statement	16
urgent we are trying to	9	i cant pick the phone	12	we are trying to contact	14
prize guaranteed call NNNNNNNNNN from	8	hope you are having a	9	prize guaranteed call NNNNNNNNNN from	13
bonus caller prize on NN	7	text me when you re	9	you have won a guaranteed	13
draw txt music to NNNNN	7	£ NNNN cash or a	8	a NNNN prize guaranteed call	12
prize N claim is easy	7	NNN anytime any network mins	8	draw shows that you have	12
you have won a guaranteed	7	a £ NNNN prize guaranteed	7	i cant pick the phone	12
a N NNN bonus caller	6	have a secret admirer who	7	urgent we are trying to	11
are selected to receive a	6	u have a secret admirer	7	call NNNNNNNNNN from land line	10

its frequency from **ADD** to **FINAL**. These 5-grams correspond to templates often present in cell phones, and used in legitimate messages. Actually, both are complete messages themselves.

- The behavior of the rest of 5-grams, which all actually nearly only occur in spam messages, is a bit different. Most of them are fuzzy duplicates that result in small frequency increases, like in "we are trying to contact" from **INIT** (10 messages) to **FINAL** (14 messages). This means that the messages in **ADD** may be duplicates of the messages in **INIT**. However, as it can be seen, the patterns of spam 5-grams within each sub-collection are very regular and even overlapping, so this is not significant. In other words, these 4 messages are not repeated, but new instances of spam probably sent by the same organization. Other messages just disappear from the top, as they keep their frequencies.

Regarding 6-grams, shown in Table V, that is the standard value used in tools like WCopyfind, we can see that the behavior is quite similar to the case of 5-grams. There are slightly different results because of two reasons:

- The fact that longer N-grams must obviously lead to lower frequencies. Actually, there is not a significant drop in the number of matches per 6-gram, as it can be seen in *e.g.* "private your NNNN account statement for", which includes the 5-gram "private your NNNN account statement" as a prefix.
- The most frequent 6-grams keep on belonging to spam messages. The 5-grams that frequently occurred on the legitimate messages have disappeared because the detected templates are, in fact, complete 5-length messages.

In 6-gram results, we can see again that there are not significant near-duplicates except for those already present in each sub-collection. Moreover, the results of 10-grams (not presented here due to space limit) are very similar to these previous ones. In consequence, we believe it is safe to say that merging the sub-collections, although they have roughly the same sources, does not lead to near-duplicates that may ease the task of detecting SMS spam.

IV. Conclusions and Future Work

In this paper, we have performed a careful analysis of the new SMS Spam Collection, which has been built in order to promote the experimentation with machine learning SMS spam classifiers. This collection has been developed by enriching a previously existing SMS corpus, using the same data sources. As a consequence, the added messages may contain previously existing messages in the original collection. Thus, it is required to ensure that this does not happen, as it may ease the task of learning SMS spam classifiers.

We have performed a detailed analysis of potential near-duplicates in the collection, by using an standard "String-to-text" method, on three sub-collections: the original one (**INIT**), the added messages (**ADD**), and the final collection (**FINAL**). The near-duplicate detection method consists of finding N-gram matches between messages, for N = 5, 6 and 10 within each collection, in order to verify that there is not a significant number of near-duplicates in the **FINAL** sub-collection, apart from those previously existing in the **INIT** and the **ADD** sub-collections.

We have found 5-grams already presented in the **INIT** and the **ADD** sub-collections do not collapse to greatly increase their frequencies, and they typically correspond to templates often presented in cell phones, and used in legitimate messages (*e.g.* "sorry i ll call later"). The 5-grams that co-occur in **INIT** and **ADD**, so they get their frequencies increased in **FINAL**, are new instances of spam most likely sent by the same organization. In 6-grams results, we have found that there are not significant near-duplicates except for those already presented in each sub-collection. Moreover, the results achieved with 10-grams are very similar to the 5- and 6-grams ones.

Table V: Ten top 6-grams and their frequencies in the studied sub-collections.

INIT		ADD		FINAL	
6-gram	#f	6-gram	#f	6-gram	#f
this is the Nnd attempt to	9	private your NNNN account statement for	15	private your NNNN account statement for	16
urgent we are trying to contact	9	i cant pick the phone right	12	a NNNN prize guaranteed call NNNNNNNNNN	12
prize guaranteed call NNNNNNNNNN from land	7	a £ NNNN prize guaranteed call	7	draw shows that you have won	12
a N NNN bonus caller prize	6	have a secret admirer who is	7	i cant pick the phone right	12
bonus caller prize on NN NN	6	i am on the way to	6	prize guaranteed call NNNNNNNNNN from land	12
cash await collection sae t cs	6	pls convey my birthday wishes to	6	urgent we are trying to contact	11
tone N ur mob every week	6	u have a secret admirer who	6	call our customer service representative on	10
you have won a guaranteed NNNN	6	£ NNN cash every wk txt	5	this is the Nnd attempt to	9
a NNNN prize guaranteed call NNNNNNNNNN	5	as i entered my cabin my	5	tone N ur mob every week	9
call NNNNNNNNNN now only NNp per	5	goodmorning today i am late for	5	we are trying to contact u	9

In consequence, we believe it is safe to say that merging the sub-collections, although they have roughly the same sources, does not lead to near-duplicates that may ease the task of detecting SMS spam.

As a future work, we plan to perform throughout experiments with machine learning content based classifiers in order to confirm and improve previous work by we and others ([9], [10], and [11]) on the much smaller SMS Spam Corpus.

ACKNOWLEDGMENT

The authors would like to thank the financial support of Brazilian agencies FAPESP, Capes and CNPq.

REFERENCES

[1] G. Cormack, "Email Spam Filtering: A Systematic Review," *Foundations and Trends in Information Retrieval*, vol. 1, no. 4, pp. 335–455, 2008.

[2] T. A. Almeida, A. Yamakami, and J. Almeida, "Evaluation of Approaches for Dimensionality Reduction Applied with Naive Bayes Anti-Spam Filters," in *Proceedings of the 8th IEEE International Conference on Machine Learning and Applications*, Miami, FL, USA, 2009, pp. 517–522.

[3] ——, "Filtering Spams using the Minimum Description Length Principle," in *Proceedings of the 25th ACM Symposium On Applied Computing*, Sierre, Switzerland, 2010, pp. 1856–1860.

[4] ——, "Probabilistic Anti-Spam Filtering with Dimensionality Reduction," in *Proceedings of the 25th ACM Symposium On Applied Computing*, Sierre, Switzerland, 2010, pp. 1804–1808.

[5] T. A. Almeida and A. Yamakami, "Content-Based Spam Filtering," in *Proceedings of the 23rd IEEE International Joint Conference on Neural Networks*, Barcelona, Spain, 2010, pp. 1–7.

[6] T. A. Almeida, J. Almeida, and A. Yamakami, "Spam Filtering: How the Dimensionality Reduction Affects the Accuracy of Naive Bayes Classifiers," *Journal of Internet Services and Applications*, vol. 1, no. 3, pp. 183–200, 2011.

[7] T. A. Almeida and A. Yamakami, "Facing the Spammers: A Very Effective Approach to Avoid Junk E-mails," *Expert Systems with Applications*, vol. 39, pp. 6557–6561, 2012.

[8] T. Almeida, J. Gómez Hidalgo, and A. Yamakami, "Contributions to the Study of SMS Spam Filtering: New Collection and Results," in *Proceedings of the 2011 ACM Symposium on Document Engineering*, Mountain View, CA, USA, 2011, pp. 259–262.

[9] G. V. Cormack, J. M. Gómez Hidalgo, and E. Puertas Sanz, "Feature Engineering for Mobile (SMS) Spam Filtering," in *Proceedings of the 30th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, New York, NY, USA, 2007, pp. 871–872.

[10] ——, "Spam Filtering for Short Messages," in *Proceedings of the 16th ACM Conference on Conference on information and Knowledge Management*, Lisbon, Portugal, 2007, pp. 313–320.

[11] J. M. Gómez Hidalgo, G. Cajigas Bringas, E. Puertas Sanz, and F. Carrero García, "Content Based SMS Spam Filtering," in *Proceedings of the 2006 ACM Symposium on Document Engineering*, Amsterdam, The Netherlands, 2006, pp. 107–114.

[12] J. P. Kumar and P. Govindarajulu, "Duplicate and near duplicate documents detection: A review," *European Journal of Scientific Research*, vol. 32, pp. 514–527, 2009.

[13] A. M. El Tahir Ali, H. M. Dahwa Abdulla, and V. Snasel, "Survey of Plagiarism Detection Methods," in *Proceedings of the 5th Asia Modelling Symposium*, Manila, Philippines, 2011, pp. 39–42.