iJRASET
International Journal For Research in
Applied Science and Engineering Technology

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Volume: 8 **Issue:** III **Month of publication:** March 2020

DOI:

International Journal for Research in Applied Science & Engineering Technology (IJRASET)
ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.429
Volume 8 Issue III Mar 2020- Available at www.ijraset.com

Sanctions Screening Tool

Swapnil Bhasgauri[1], Tejas Kulkarni[2], Srisail Madduru[3], Mrinal Khadse[4], Niraj Borkar[5]
[1, 2, 3, 4, 5]Department of Information Technology, SIES Graduate School of Technology

Abstract: The US and the European associations have levied economic sanctions on nations, individuals and business entities that are classified as either degenerate, nefarious, shell organizations engaged with tax evasion and illegal financing for individuals or organizations enjoying illicit activities. Because of these, financial institutions are required to abstain from transacting with individuals and entities situated in these sanctioned countries and also enabling sanctioned elements to transact on their platform. So as follow these guidelines, all such establishments are required to screen each exchange and clients on their platform and report people, organizations and transactions from sanctioned countries to the regulators. Inability to do so can result in extreme fines.

We propose to build up a cloud based application that uses algorithms, for example, Levenshtein algorithm. The input to this can be dialects apart from English as this system would have the ability to translate or transliterate (depending upon the language) from a frontier language to English. This system is going for decreasing false positive alerts and subsequently would utilize a feedback circle to diminish false positive alerts.

Keywords: Levenshtein algorithm, pattern matching, sanctions screening, string matching, KMP algorithm

I. INTRODUCTION

Since September 11 2001, there has been a mentioned shift towards focused or so-referred to as smart sanctions, which aims to minimize the suffering of harmless civilians. Sanctions take a variety of bureaucracy, which includes travel bans, asset freezes, hands embargoes, capital restraints, overseas resource reductions, and change restrictions.[1]

International sanctions are political and monetary decisions which are a part of diplomatic efforts via nations, multilateral or local companies in opposition to states or companies either to shield national protection interests, or to shield worldwide law, and protect towards threats to worldwide peace and security.

These decisions principally encompass the temporary imposition on a goal of economic, exchange, diplomatic, cultural or different restrictions (sanctions measures) which are lifted whilst the motivating safety issues no longer practice, or whilst no new threats have arisen.[2]

National governments and worldwide bodies which include the United Nations and European Union have imposed monetary sanctions to coerce, deter, punish, or disgrace entities that endanger their hobbies or violate international norms of conduct. Sanctions have been used to enhance more than a few overseas coverage dreams, which include counterterrorism, counternarcotics, nonproliferation, democracy and human rights advertising, battle resolution, and cybersecurity. [3]

There are many such tools which produce similar results but with false positives. Our idea helps reducing such false positives and subsequently would utilize a feedback circle to diminish false positive alerts.

II. NEED AND CHALLENGES

All corporations are obliged to comply with sanctions screening requirements, and consequently need to have ok controls in region. Historically, enforcement movements had been greater distinguished in Financial Services, however other sectors have additionally received sizeable fines and a few regulatory our bodies are increasingly turning their interest to other industries. For example, the Office of Financial Sanctions Implementation (OFSI) has posted monetary sanctions steerage for charities and non-governmental firms.

Additionally, the US OFAC has issued massive fines to enterprises throughout a extensive variety of sectors out of doors of the United States for now not having conducted suitable sanctions exams. View this manual to US OFAC sanctions for extra detail on whether or not your business is in scope and the way not to breach OFAC sanctions.

The latest entity which was fined for transacting with a sanctioned entity or a country was fined an amount of 1,125,000 $ which is to be paid in one settlement. To prevent this sanction screening tool are a very big requirement right now for every individual or a company.

The real mission for lots businesses is not just to locate clients who are on sanctions lists and prevent them from transacting with the enterprise, but also to avoid disrupting the consumer journey for legitimate customers and undermining the performance of the corporation's operations.

1) *Under or Over Screening:* If organisations do not screen robustly, there is a chance of 'false negatives', wherein entities challenge to sanctions slip via the net.Conversely, over-screening can bring about firms generating high volumes of 'false positives', wherein non-sanctioned entities are flagged as probably sanctioned. These false positives need time and resource to remediate to verify they are not sanctioned.[3]
2) *Equivalence:* Whilst formerly commonplace, counting on a 3rd birthday party for sanctions compliance or 'equivalence' is now not applicable. For example, banks historically trusted the sanctions screening controls in their correspondent banks for mutual clients**.** This is now not permissible.[3]
3) *Divergence:* In sure instances, the economic sanctions carried out by exceptional sanctioning our bodies are inconsistent. For example, with Iranian sanctions, OFAC and the EU have a one-of-a-kind stance – OFAC has decided to reinstate sanctions against Iran, whilst the EU is still offering sanctions relief and encouraging EU groups to have interaction with Iran.When transacting with an entity sanctioned by one frame but no longer every other, you have to exhibit more warning and enforce additional controls.[3]

III.LITERATURE SURVEY

Most organizations have a system currently, but they generate a lot of false positive alerts causing to a significant monetary loss. Financial institutions are scaling and entering new markets wherein they are using regional dialect in order to enhance user experience. However, the existing systems are able to screen in English only which then requires the institutions to translate/ transliterate from native language to English. The third challenge that these financial institutions face is that the existing on-premise systems are expensive to install and maintain on an on-going basis, specially for start-up institutions. A alternate solution to this is to build a cloud based solution.

We propose to develop a cloud based solution that leverages algorithms for example Levenshtein distance as the base. The input to this solution can be languages apart from English as this solution would provide the functionality to translate or transliterate (depending on the language) from a colloquial language to English. This solution is aimed at reducing false positive alerts and hence would use a feedback loop based on machine learning techniques to reduce false positive alerts.

The Levenshtein distance is a string metric for measuring the difference among sequences. Informally, the Levenshtein distance among two words is the minimum wide variety of individual edits (i.E. insertions, deletions, or substitutions) required to trade one word into the opposite.[4]

Set n to be the length of s.
Set m to be the length of t.
If n = 0, return m and exit.
If m = 0, return n and exit.
Construct a matrix containing 0..m rows and 0..n columns.
Initialize the first row to 0..n.
Initialize the first column to 0..m.
Examine each character of s (i from 1 to n).
Examine each character of t (j from 1 to m).
If s[i] equals t[j], the cost is 0.
If s[i] doesn't equal t[j], the cost is 1.
Set cell d[i,j] of the matrix equal to the minimum of:
The cell immediately above plus 1: d[i-1,j] + 1.
The cell immediately to the left plus 1: d[i,j-1] + 1.
The cell diagonally above and to the left plus the cost: d[i-1,j-1] + cost.
After the iteration steps (3, 4, 5, 6) are complete, the distance is found in cell d[n,m].

A string-matching set of rules wants to locate the starting index m in string S[] that matches the hunt word W[].

The most truthful set of rules, referred to as the "Brute-pressure" or "Naive" algorithm, is to look for a phrase suit at each index m, the position inside the string being searched, i.E. S[m]. At each role m the set of rules first checks for equality of the primary character in the word being searched, i.E. S[m] =? W[0]. If a in shape is observed, the algorithm tests the alternative characters

International Journal for Research in Applied Science & Engineering Technology (IJRASET)
ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.429
Volume 8 Issue III Mar 2020- Available at www.ijraset.com

within the phrase being searched by means of checking successive values of the word role index, i. The algorithm retrieves the character W[i] within the word being searched and assessments for equality of the expression S[m+i] =? W[i]. If all successive characters fit in W at role m, then a in shape is found at that role within the seek string. If the index m reaches the cease of the string then there may be no suit, wherein case the hunt is stated to "fail".

Usually, the trial check will fast reject the trial healthy. If the strings are uniformly dispensed random letters, then the hazard that characters healthy is 1 in 26. In maximum instances, the trial test will reject the in shape on the initial letter. The chance that the first letters will healthy is 1 in 262 (1 in 676). So if the characters are random, then the expected complexity of looking string S[] of duration okay is on the order of okay comparisons or O(k). The anticipated overall performance is excellent. If S[] is 1 million characters and W[] is 1000 characters, then the string search ought to complete after approximately 1.04 million man or woman comparisons.

That predicted overall performance is not assured. If the strings are not random, then checking a trial m may additionally take many character comparisons. The worst case is if the two strings suit in all but the ultimate letter. Imagine that the string S[] consists of 1 million characters which are all A, and that the word W[] is 999 A characters terminating in a final B person. The simple string-matching algorithm will now examine one thousand characters at every trial position before rejecting the suit and advancing the trial role. The simple string search instance might now take approximately a thousand man or woman comparisons instances 1 million positions for 1 billion man or woman comparisons. If the period of W[] is n, then the worst-case performance is O(ok·n).[4-9]

The KMP algorithm has a higher worst-case overall performance than the straightforward algorithm. KMP spends a touch time precomputing a desk (on the order of the scale of W[], O(n)), and then it uses that table to do an efficient seek of the string in O(ok).[10]

```
def KMPSearch(pat, txt):
    M = len(pat)
    N = len(txt)

    # create lps[] that will hold the longest prefix suffix
    # values for pattern
    lps = [0]*M
    j = 0 # index for pat[]

    # Preprocess the pattern (calculate lps[] array)
    computeLPSArray(pat, M, lps)

    i = 0 # index for txt[]
    while i < N:
        if pat[j] == txt[i]:
            i += 1
            j += 1

        if j == M:
            print "Found pattern at index " + str(i-j)
            j = lps[j-1]

        # mismatch after j matches
        elif i < N and pat[j] != txt[i]:
            # Do not match lps[0..lps[j-1]] characters,
            # they will match anyway
            if j != 0:
                j = lps[j-1]
            else:
                i += 1
```

```
def computeLPSArray(pat, M, lps):
    len = 0 # length of the previous longest prefix suffix

    lps[0] # lps[0] is always 0
    i = 1

    # the loop calculates lps[i] for i = 1 to M-1
    while i < M:
        if pat[i]== pat[len]:
            len += 1
            lps[i] = len
            i += 1
        else:
            # This is tricky. Consider the example.
            # AAACAAAA and i = 7. The idea is similar
            # to search step.
            if len != 0:
                len = lps[len-1]

                # Also, note that we do not increment i here
            else:
                lps[i] = 0
                i += 1

txt = "ABABDABACDABABCABAB"
pat = "ABABCABAB"
KMPSearch(pat, txt)
```

The KMP algorithm has a lower efficiency than Levenshtein algorithm has hence we use the latter for implementation.

IV. METHODOLOGY

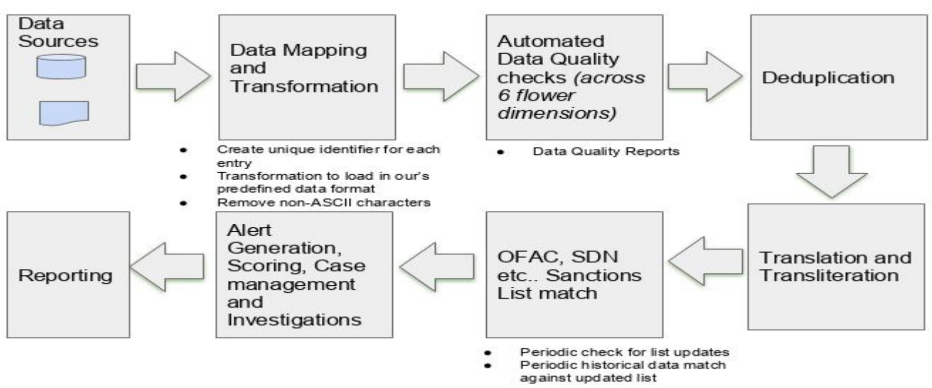


Figure 1: Flow Diagram

1) *Step 1:* Data Sources: Data sources consists of current unscreened transactions as well as a list of all the sanctioned entities to which our transactions will be compared to.[8]
2) *Step 2:* Data Mapping and transformation: The unscreened transactional data will be transformed to a predefined data format. It also removes all the non ASCII characters in order to make the screening more efficient.
3) *Step 3:* Automated data Quality Checks: Data quality reports are to be generated periodically
4) *Step 4:* Deduplication : Redundant data can be removed using various libraries.

5) *Step 5:* Translation and transliteration : This step includes the literal translation of the entity title as well as the other data provided in the transactions data set.[11][12]
6) *Step 6:* Sanction List match: Our sanctions data list will be periodically updated with the OFAC,SDN etc to keep up with the government sanctions. In addition to returning consequences which can be actual matches , Sanctions List Search also can offer a broader set of results using fuzzy common sense. This common sense makes use of person and string matching in addition to phonetic matching. Only the name discipline of Sanctions List Search invokes fuzzy common sense when the tool is run. The other fields on the device use person matching good judgment.
7) *Step 7:* Scoring: The score subject indicates the similarity between the name entered and resulting matches on one of OFAC's sanctions lists. It is calculated with the use of matching logic algorithms: one primarily based upon phonetics, and a second based upon the similarity of the characters inside the two strings. A rating of one hundred indicates an actual in shape, whilst dec0.0.0.rease rankings suggest capability matches.[13-15]
8) *Step 8 :* Reporting: A report is generated which includes all the details about the whole screening process as well as the sanctioned entities

V. RESULTS AND DISCUSSIONS

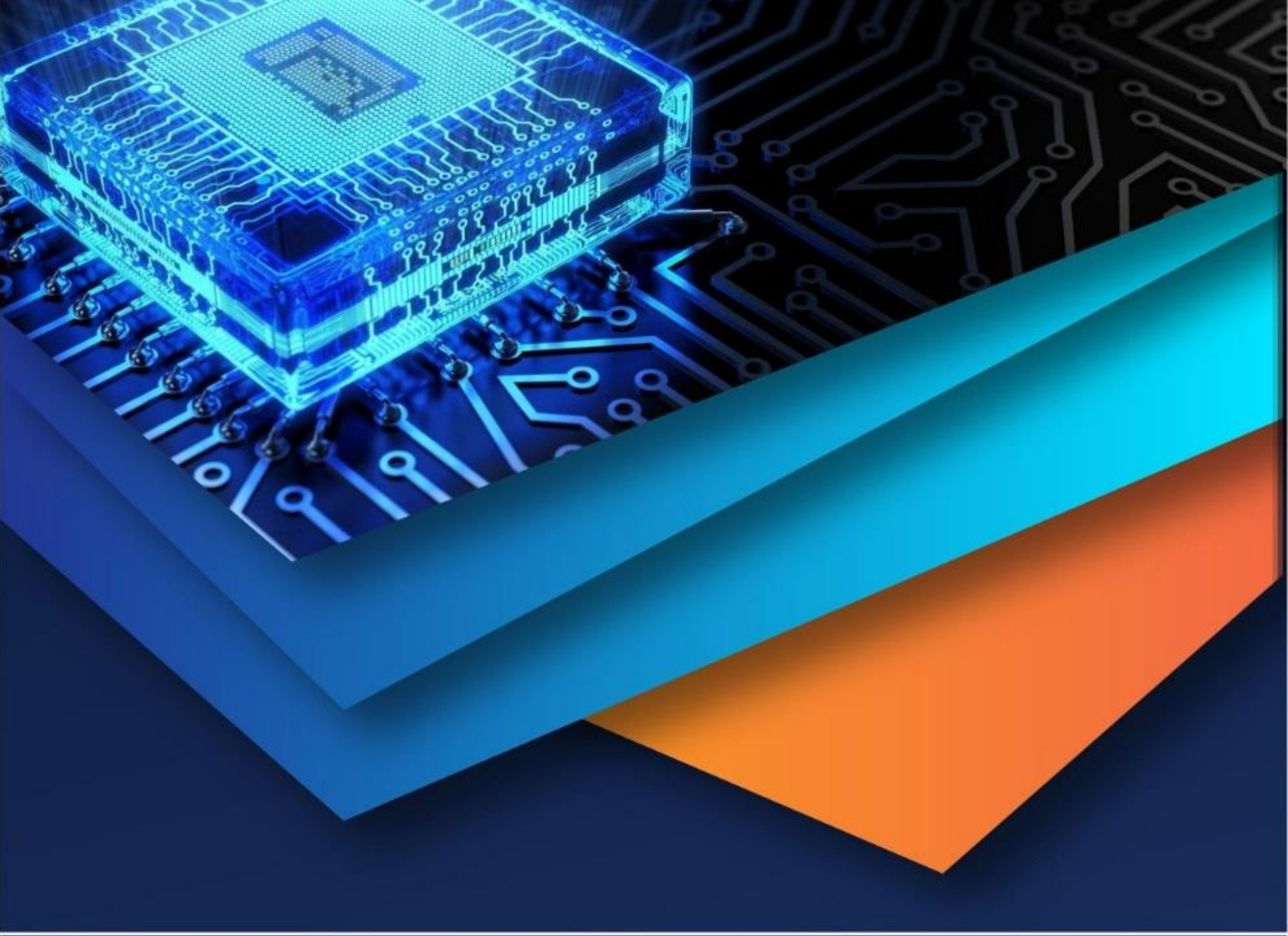
INPUT	Names Screened	Score
Sw@pn!l	Swapnil	70
Niraj	N!r@j	60
	Niraj Kathkar	100
	neeraj	80

Table 1: Searching name of the entity

This is how the procedure takes place. Suppose we are to screen a sheet of transactions and look out for an individual named Swapnil. To get away with it the aliases used may be in lines of "Sw@pn!l"etc. Our methodology successfully scores and screens out all such possibilities as above and thus can improve the rate of detection and lowers the number of false positives prevailing in the procedure.

REFERENCES

[1] R. Centeno, H. Billhardt and R. Hermoso, "An Adaptive Sanctioning Mechanism for Open Multi-agent Systems Regulated by Norms," 2011 IEEE 23rd International Conference on Tools with Artificial Intelligence, Boca Raton, FL, 2011, pp. 523-530.
[2] M. R. Robert, M. I. Cortés, G. A. L. Campos and E. S. S. Freire, "A sanction-application mechanism considering commitment levels in hierarchical organizations," 2015 Latin American Computing Conference (CLEI), Arequipa, 2015, pp. 1-11.
[3] Luis G. Nardin, Tina Balke-Visser, Nirav Ajmeri, Anup K. Kalia, Jaime S. Sichman, Munindar P. Singh, "Classifying sanctions and designing a conceptual sanctioning process model for socio-technical systems", The Knowledge Engineering Review, vol. 31, pp. 142, 2016
[4] S. Zhang, Y. Hu and G. Bian, "Research on string similarity algorithm based on Levenshtein Distance," 2017 IEEE 2nd Advanced Information Technology, Electronic and Automation Control Conference (IAEAC), Chongqing, 2017, pp. 2247-2251.
[5] Y. Zhou and G. Ding, "Research of a Pattern Matching Algorithm Based on Threshold and Word Frequency," 2018 IEEE International Conference on Computer and Communication Engineering Technology (CCET), Beijing, 2018, pp. 320-324
[6] P. Zhan, W. Yuping and X. Jinfeng, "An Improved Multi-pattern Matching Algorithm for Large-Scale Pattern Sets," 2014 Tenth International Conference on Computational Intelligence and Security, Kunming, 2014, pp. 197-200.
[7] D. Zhao, X. Zhu and T. Xu, "Improvement of algorithm for pattern matching in intrusion detection," 2013 5th IEEE International Conference on Broadband Network & Multimedia Technology, Guilin, 2013, pp. 281-284.
[8] P. Venkata Subba Reddy, "Generalized fuzzy logic for incomplete information," 2013 IEEE International Conference on Fuzzy Systems (FUZZ-IEEE), Hyderabad, 2013, pp. 1-6.doi: 10.1109/FUZZ-IEEE.2013.6622305
[9] L. Deng, Y. Cai, C. Wang and Y. Jiang, "Fuzzy Temporal Logic on Fuzzy Temporal Constraint Networks," 2009 Sixth International Conference on Fuzzy Systems and Knowledge Discovery, Tianjin, 2009, pp. 272-276..
[10] U. S. Alzoabi, N. M. Alosaimi, A. S. Bedaiwi and A. M. Alabdullatif, "Parallelization of KMP string matching algorithm," 2013 World Congress on Computer and Information Technology (WCCIT), Sousse, 2013, pp. 1-3
[11] M. A. Zahid, N. I. Rao and A. M. Siddiqui, "English to Urdu transliteration: An application of Soundex algorithm," 2010 International Conference on Information and Emerging Technologies, Karachi, 2010, pp. 1-5.
[12] W. Xiong and Y. Jin, "A new Chinese-English machine translation method based on rule for claims sentence of Chinese patent," 2011 7th International Conference on Natural Language Processing and Knowledge Engineering, Tokushima, 2011, pp. 378-381
[13] Weisong Chen and Liang Shi, "Credit scoring with F-score based on support vector machine," Proceedings 2013 International Conference on Mechatronic Sciences, Electric Engineering and Computer (MEC), Shengyang, 2013, pp. 1512-1516.
[14] J. Shi, S. Y. Zhang, and L. M. Qiu, "Credit scoring by feature-weighted support vector machines," Journal of Zhejiang University, Vol. 30, no. 4, pp. 994-996, 2004.
[15] P. Yao, "Feature selection based on svm for credit scoring," in 2009 International Conference on Computational Intelligence and Natural Computing, pp. 46-48, 2009



10.22214/IJRASET

INDEX COPERNICUS
45.98

IMPACT FACTOR:
7.129

TOGETHER WE REACH THE GOAL
IMPACT FACTOR:
7.429

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY