FACTU: AN ANDROID APPLICATION FOR NEW VERIFICATION USING SENTENCE SIMILARITY ANALYSIS AND HIDDEN MARKOV MODEL

EMMANUEL B. CONSTANTINO JR.
HOMER C. MALIJAN

SUBMITTED TO THE FACULTY OF INSTITUTE OF COMPUTER SCIENCE
UNIVERSITY OF THE PHILIPPINES LOS BAÑOS
IN PARTIAL FULFILLMENT OF THE
REQUIREMENTS FOR THE
DEGREE OF

BACHELOR OF SCIENCE
(Computer Science)

JUNE 2017

The special problem hereto attached entitled FACTU: AN ANDROID APPLICATION FOR NEW VERIFICATION USING SENTENCE SIMILARITY ANALYSIS AND HIDDEN MARKOV MODEL, prepared and submitted by EMMANUEL B. CONSTANTINO JR. AND HOMER C. MALIJAN, in partial fulfillment of the requirements for the degree of BACHELOR OF SCIENCE (Computer Science) is hereby accepted.

Accepted in partial fulfillment of the requirements for the degree of BACHELOR OF SCIENCE (Computer Science)

Caroline Natalie Peralta
Adviser

Date Signed

Jaime Samaniego
Director, Institute of Computer Science

Date Signed

BIOGRAPHICAL SKETCH

One of the authors, Homer C. Malijan, was born on January 17, 1997 in Sta. Cruz, Laguna but is currently living in Calauan, Laguna as the eldest child of Janette C. Malijan and Raul L. Malijan.

He was enroll in The Refiner's Christian School in Calauan, Laguna as an elementary student. He took his secondary education at Pedro Guevara Memorial National High School Special Science Curriculum and is consistent to be on the top section. Homer is not only academically inclined but is also a fitness enthusiast, he enjoys playing Football and Basketball. In 2013, he was admitted to the University of the Philippines Los Baños under the Bachelor of Science in Computer Science program of the Institute of Computer Science.

HOMER CORTEZ MALIJAN

Emmanuel B. Constantino Jr. was born on November 19, 1996 in Albay, Bicol. He is the youngest son of Ma Victoria Constantino and Emmanuel Constantino. He was enrolled in Joy in Learning School from Nursery to High School. He loves to play computer games and sports.

EMMANUEL BENITO CONSTANTINO JR.

ACKNOWLEDGEMENTS

I would like to thank Professor Caroline Natalie Peralta for her expert advice throughout this project, as well as my colleague, Homer Cortez Malijan, for this wonderful collaboration and his splendid cooperation. I would like to thank my family for the emotional support that they have provided me. I express my deepest gratitude towards my closest friends, JC Carlo Quintos, Diego Sevilla, Gabrielle Pauline Payawal, Michael Jowell Lapie, and Kenji Mogi for always being there in the toughest times in creating this project. To my organizations, Alliance of Computer Science Students, Infocus Multimedia Guild and UP Epsilon Chi Fraternity, I would like to thank you all for being my 2nd family in my stay here in the university. And last but not the least, I would like to thank God for the unending grace, provision and courage to face the trials that I have encountered in doing this project.

-MJ

I would like to express my appreciation and indebtedness particularly to the following:

My Adviser, Ms. Caroline Natalie Peralta, and my colleague, Emmanuel Constantino Jr., for the time and encouragement they have provided me during the course of this project.

My parents, Raul and Janette Malijan, who shared their endless support, either morally, financially, or physically, thank you.

I would also want to extend my gratitude to my peers, Carol Anne Marie Adajar, Jose Diego Sevilla, and JC Carlo Quintos for their continuous motivation for us to finish this study.

Above all, I would like to thank the Great Almighty God, the author of knowledge and wisdom, for His guidance.

-Homer

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

ABSTRACT

EMMANUEL B. CONSTANTINO JR. AND HOMER C. MALIJAN, University of the Philippines Los Baños, JUNE 2017. FACTU: AN ANDROID APPLICATION FOR NEWS VERIFICATION USING SENTENCE SIMILARITY ANALYSIS AND HIDDEN MARKOV MODEL. Major Professor: PROF. CAROLINE NATALIE PERALTA

The Internet's ecosystem grows every second, and with it comes new information that might not be reliable or accurate. One of the biggest problems people face when surfing the Internet is information fabrication. Although possibly unintentional, the need to validate such statements is necessary. This paper presents an application that crawls through various reliable and satiric websites for information and matches statements to assess its validity. With over 5,000 news articles, a Hidden Markov Model (HMM) was used to structure relevant information depending on the user's input to be evaluated for its category: verified, satiric, or unverified.

INTRODUCTION

Background of the Study

As the literacy rate of the world increases, verification of information will be the next concern. The world is also starting to appreciate the World Wide Web, making it easier for people of any age bracket to surf for information on the Internet. This causes the Internet to be prone to hackers, trolls, or fake online information, since most users believe what they see on the Internet.

The study aims to verify a statement's validity by crawling through trusted and “satiric" web pages and comparing relevant articles, rating its resemblance and likeness, and use a Hidden Markov Model (HMM) to classify such input.

Significance of the Study

The inevitable growth of Internet users has made it a lot easier for information to spread in a short span of time. Not all information that is produced online can be reliable since all Internet users have the capability of reproducing and spreading information especially in social media websites. This results in plenty of people acquiring questionable information which makes it harder for them to determine which information is legitimate. The study is relevant because of the increase in the number of hoax websites disseminating misleading information, especially in the field of politics and science.

Objective of the Study

The main purpose of the study is to develop an application that crawls the Internet for relevant information as reference to classify input statements as verified, unverified, or satiric.

Specifically, the study aims to achieve the following:

1. To develop an Android application using Python that can help people verify statements that they see online using their smartphones, the same way dictionary applications are used to verify meaning of certain words;
2. To web crawl various news websites to feed the database with information to better assess the user-defined statement;
3. To evaluate whether or not a news article supports a particular statement using sentence similarity analysis; and
4. To assess the effectiveness of using an HMM to classify news statements.

Scope and Limitation of the Study

This study will focus on verifying news by analyzing data that was obtained from different trusted and satirical news websites. The range of the news from trusted news websites will be from local to international news. The information that will be gathered from these news websites will be evaluated through Sentence Similarity Analysis. The scores computed from previous analysis will be used to feed the HMM that will produce the final output.

The computational model will greatly depend on the information gathered from various news websites. In essence, if a satirical website posts genuine news, the application will still label it as satiric because of its source.

The application can only process statements that are not compound-complex sentences in structure to avoid inputs having multiple news content. Moreover, news websites will only be crawled every 24 hours, meaning news may not be available as a source as soon as it is published. Furthermore, this study is limited to news articles written in English.

REVIEW OF RELATED LITERATURE

Sentence Similarity Analysis

One of the crucial parts of this study is comparing various sentences and evaluating its similarity. There are many ways to measure the similarity between sentences. One method is through Word Overlapping, this is based on computing the number of exact words that can be seen in each sentence. Another method is through Linguistic Analysis. This method can be implemented in three different ways (Achananuparp, n.d.):

1. **Semantic Textual Similarity** combines Latent Semantic Analysis and Machine Learning algorithms.
2. **Word Order Similarity** focuses on the measurement of basic syntactic information; the permutation of words.
3. **Hybrid of Semantic and Syntactic Similarity Analysis**

Grammar Approach is another way in which the calculation can be done, based on a grammatical structure derived from parsing sentences (Lee, 2014).

There are plenty of related studies in the field of Sentence Similarity Analysis. There exist a research conducted for exploring methods in creating an Automated Short-Essay grading application which evaluates the students' answers based on the correct answer given by the teacher.

Measuring similarity is subjective but it can also be computed using different methods:

1. **Euclidean Distance** is one of the most common metrics for measuring proximity; the closer the distance, the higher the similarity.

2. **Manhattan Distance** is measured by getting the total sum of the difference between x-coordinates and y-coordinates.
3. **Minkowski Distance** is a generalized method of getting distance. Both Euclidean and Manhattan Distance can be derived from it.
4. **Jaccard Similarity** is a method where objects are considered as sets. It simply gets the cardinality of the intersection of the sets divided by the cardinality of the union of the sets.
5. **Cosine Similarity** finds the cosine of the angle between the two objects (see Fig. 1). It sees objects as vectors and computes for the normalized dot product. This similarity measure is used in this study.

$$\textbf{similarity} = \frac{\sum_{i=1}^{n} A_i B_i}{\sqrt{\sum_{i=1}^{n} A_i^2}\sqrt{\sum_{i=1}^{n} B_i^2}}$$

Fig. 1. Formula of Cosine Similarity

ASOBEK system is a software that aides in Sentence Similarity Analysis which uses the SVM classification algorithm with the Word Overlapping method. On the other hand, the MITRE system uses Neural Networks with string matching features. Studies in the field of Sentence Similarity Analysis have greatly progressed in the past few years due to the increasing development of Artificial Intelligence. The growth can also be attributed to an annual competition named SemEval (Kashya, 2015), where developers create systems that use Sentence Similarity Analysis based on a particular theme.

Web Crawler

Web Crawling is an application of Artificial Intelligence which is usually used in applications that require a wide array of data gathering procedures. These crawlers are typically called spiders that are deployed in web pages that functions as a robot that visits other links normally done recursively to exhaust all the links that may contain information about the topic of interest. There are various approaches prior to this study that focuses on the advancement of Web Crawling such as those discussed by Rahul Kamar, Anurag Jain, and Chetan Agrawal in their journal article on Survey of Web Crawling Algorithms (Kumar, 2014) including Breadth First Search, Depth First Search, Page Rank Algorithm, Crawling Through URL Ordering, Batch-page Rank, Partial-page Rank, Using HTTP Get Request and Dynamic Web Pages, etc. (Olston, 2010). All this algorithms are used to further improve the performance of spiders because they usually run tedious processes and allocates a huge amount of information into a database for later computation.

Since the Internet is a vast place, diverse algorithms have been developed to further amplify its performance like skipping irrelevant web pages (Manku, 2007) as described by Gurmeet Manku, Arvind Jain, and Anish Sarma which greatly affected a web crawler's run time, ignoring ads and unrelated and malicious links. However, this technique is still in development because of the subjectivity in considering a link to be irrelevant to the topic of interest. Another factor that affects a web crawler's performance is keyword extraction as explained by Gunjan Agre and Nikita Mahajan that hastens a crawler's response (Agre, 2015) as it does not require a relevant feedback before it is considered to be a part of the training data since the crawler matches keywords from a certain topic with the keywords predefined within various web pages.

Some crawlers that require parsing more than the Surface Web have also been developed by different programmers like that of Xiang Peisu, Tien Ke, and Huang Qinzhen that developed an effective framework for Deep Web Crawling (Peisu, 2008) that makes use of a number of novel techniques to scrape information from the Deep Web.

Python and Android

Python has been one of the most popular programming languages since its development that started in the late 1980's by Guido van Rossum, at CWI in Netherlands, primarily as a successor of the ABC programming language. (Tulchak, n.d.) Nowadays, Python is known to be an open source, general purpose scripting language widely used and has a very wide array of libraries for various purposes.

On the other hand, Android programming is hastily developing due to the recent smartphone revolution. 80% of smartphones run in Android, thus, the hype to develop a huge diversity of libraries to help programmers develop their own Android applications. And because of this, many developers as young as 12 years old in different fields starts to engage in coding Android applications either for leisure or as an extra source of income since google, owner of Android, made it easier for people to sell their applications online through Google Play, a built in application that acts like a store for users to easily skim through new applications.(Kashyap, n.d.)

Due to both technology's increasing popularity, the need to develop an application that is written in Python and runs on Android grows. Libraries such as python-for-android is created to ease developers with this issue. It allows programmers to package Python code into standalone Android

Packages (APKs) that can be copied or even uploaded to different marketplaces such as the Android Play Store (Kivy, 2016).

Hidden Markov Models

HMM is one of the most popular models in Artificial Intelligence for sequential or temporal data. This model is simple yet efficient in recuperating a series of states from a series of observation (Ramage, 2013). At present, HMM is used in various speech recognition application and numerous computational problems in molecular biology (Ghahramani, 2001).

Given a finite set of discrete random variables Z, and a set of values X that takes the value of some distinct set that describes the corresponding Z variable it pairs with (see Fig. 2).

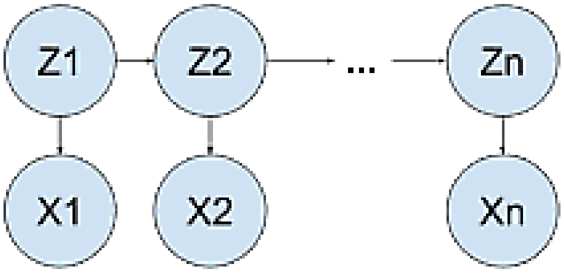


Fig. 2. Trellis Diagram

Related Works

A group of four college students recently created a Chrome browser extension for a hackathon at Princeton University in New Jersey, USA using Python 3.0 to verify links. Google search was used to query the input and check if they are trustworthy based on a set of reliable websites (Goel,2016). Their extension supports verifying twitter snap chats as well. They use Google's Tesseract OCR to extract the tweet as well as the username. They then query it online and open the profile then parse every tweet and look for a match with the text extracted from the snapchat (Bort,2016) (see Fig. 3).

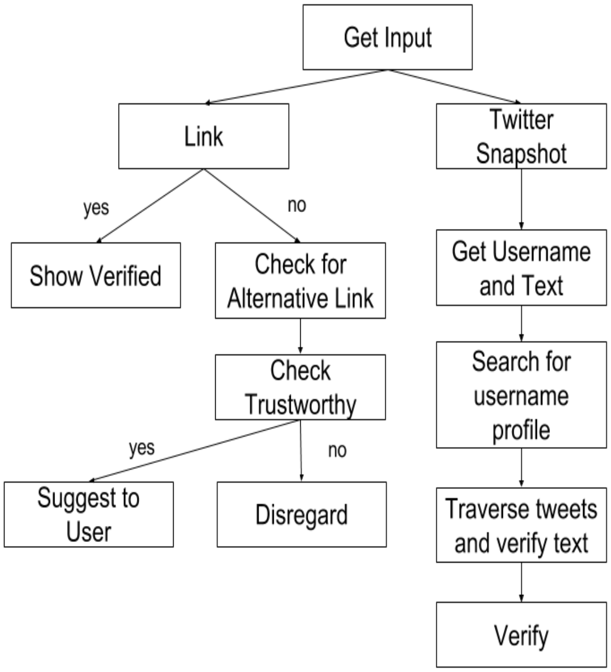


Fig. 3. Program flow of the Chrome browser extension by a group of college students

METHODOLOGY

Web Crawling and Database

The application made use of an SQLite3 database for a simpler implementation of a database in terms of Python. It also has a Graphical User Interface (GUI) that makes it easier to monitor stored data as well as migrate information since it generates a database file which can be copied from different systems.

The Web Crawler deploys a spider to the home page of every news website, both credible and satiric (see Table 1). This spider visits every link in the current page and exhausts every link recursively to try to take the title, author, publish date, body of the news, as well as the URL and stores it in the database. Some of the pages do not directly give the author and publish date of the news because some of which from the same website has a different HTML structure which in a way confuses the spider which information to store, thus, leaves the column blank to prevent processing of faulty information.

Reliable Websites	Satiric Websites
http://www.aljazeera.com/ http://www.bbc.com/news http://edition.cnn.com/ http://www.foxnews.com/ https://news.google.com/ https://www.theguardian.com http://www.nbcnews.com/	https://adobochronicles.com/ http://www.newsbiscuit.com/ Dailycurrant.com theonion.com http://www.socialnewsph.com/

Table 1 List of Reliable and Satiric Websites

A script is created which linearly deploys all these spiders and maintains the database at the same time. The script contains a loop that re-runs the same process every 24 hours to catch new information on every concerned website's home page.

Sentence Similarity Analysis

Latent Semantic Analysis is a method used for analyzing the relationship between terms and statements which will be used in retrieving similar sentences. The platform to be used to achieve such method is the Natural Language Toolkit or more commonly known as NLTK. NLTK is one of the most popular packages used in natural language processing. Under this package is the Wordnet, a lexical database which is used in determining definition of words, synonyms, homonyms, and identifying parts-of-speech.

The input along with the list of data retrieved from the server based on the input's subject are preprocessed before computing the similarity. It undergoes tokenization of words along with the parts-of-speech which are tracked word per word. BLLIP parser was also used in this study to determine the syntactic structure of the string.

After processing each word in the string, there are different approaches used to calculate the Semantic Similarity between sentences.

1. **Triplet Extraction Approach** - This approach was derived from the paper of Yuntong Liu and Yanjun Liang about calculating Sentence Semantic Similarity based on Segmented Semantic Comparison (Yuntong,2013). Triplet extraction aims to track the subject, verb, and object (SVO) of a sentence. This study focused on one of the standard and basic structures of a sentence. The idea is to get the SVO of the sentences and compare each segment to its corresponding segment in the other sentences. This approach used Cosine Similarity (finds the cosine of the angle between the two objects. It sees objects as vectors and computes for the

normalized dot product) for computing the semantic relatedness between the set of words and using the Wu-Palmer Similarity for computing the semantic relatedness between each word.

The syntactic structure produced by the parser was used to get the subject-verb-object of the sentence. The set of nouns found in the first Noun Phrase (NP) is the subject of the sentence while the deepest verb found in the Verb Phrase (VP) subtree before encountering a Noun Phrase, Prepositional Phrase (PP) or an Adjective Phrase (ADJP) is the verb of the sentence. Lastly, nouns and adjectives encountered in the last subtrees are the objects of the sentence. Proper nouns that are consecutively placed in the sentence are automatically merged. After getting the SVO of the sentences, the segments should be turned into vector spaces in order to compute the relatedness using Cosine Similarity. Each word would be compared to each other using the Wu-Palmer Similarity, which shows how similar two-word senses are based on the depth of the senses in the taxonomy. After getting each score from each segment, coefficients will be used to weigh each segment. The sum of each segment multiplied to its corresponding coefficient is the final similarity score of the 2 sentences.

2. **NV-space Approach -** This approach was implemented based on the paper of Ming Che Lee, Jia Wei Zhang, Wen Xiang Lee, and Heng Yu Ye on Sentence Similarity Computation based on Parts-of-Speech and Semantic Nets (Lee,2009). The basic idea is to get all of the nouns and verbs of the sentence to form a noun and verb set. Each set would be compared to its corresponding set in the other sentences. This approach also uses Cosine Similarity in order to get the Noun Cosine and the Verb Cosine. Wu-Palmer Similarity was also used for getting the relatedness between each word. The proposed coefficients to be used were 0.65 and 0.35 for the Noun Cosine and Verb Cosine respectively.

3. **Semantic and Word Order Similarity Approach** - Yuhua Li, Zuhair Bandar, David McLean and James O’Shea conducted a study about measuring Sentence Similarity for Short Sentences

(Li, n.d.). The main idea of the approach is to also create a vector space for both sentences and to measure the shortest path length between words to get the similarity. It also considers Word Order Similarity in different orders may result in different meanings. Coefficients were also used for weighing the Semantic Similarity and the Word Order Similarity. Adding the two similarities would result in the final similarity score.

These approaches would return similarity scores which will be used for the HMM in order to verify the input.

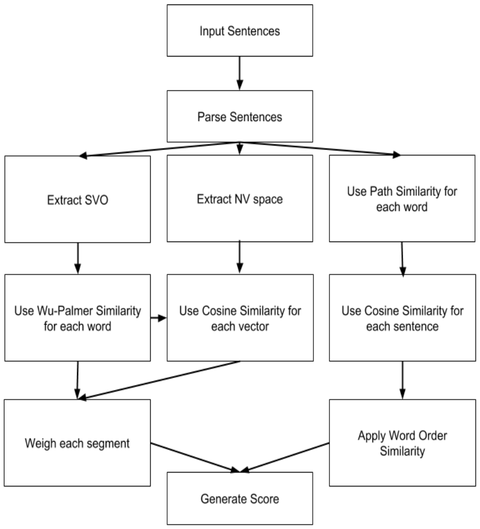

Input Sentences
Parse Sentences
Extract SVO
Extract NV space
Use Path Similarity for each word
Use Wu-Palmer Similarity for each word
Use Cosine Similarity for each vector
Use Cosine Similarity for each sentence
Weigh each segment
Apply Word Order Similarity
Generate Score

Fig. 4 Sentence Similarity Analysis flow chart

Hidden Markov Models

The model is constructed using the similarity scores received from the Sentence Similarity Analysis module as to a set of sentences, Z, formed like a Markov Chain with corresponding pairing with the type of website, X, if it came from, either reliable or satiric. First is to set up the parameters. The states in this study are Verified, Satiric and Not Verified. Mainly, there is only 2 states which is Verified and Satiric, but the state Not Verified is used when the model lacks input or evidence. The starting probability are both 0.5 for Verified and Satiric to have a fair computation for determining the last state of the query given a set of observations. The emission probability would depend on the observations. If the news came from a credible website, then the emission probability of that news being Verified is its similarity score and the complement of that score would be the emission probability of that news being Satiric. Lastly, 0.5 was the value used for the transition probability of all possible combinations. After setting up the parameters, the Viterbi Algorithm was used to compute for finding the maximum overall possible state sequence.

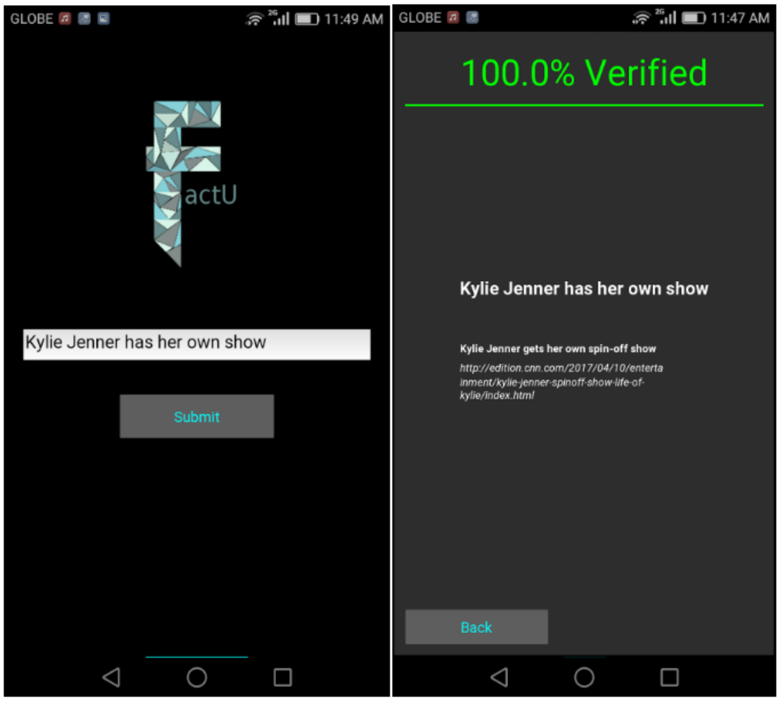
The Viterbi algorithm gets the "most-likely" sequence of hidden states given a history of evidences. With this algorithm, we can also get the 'belief state' which is the probability of a hidden state at a certain time. The most important part of the results produced by the algorithm for this application is the last 'belief state' which means that the algorithm has already traversed all the observed news events sorted according to published date.

Android Application using Python

Kivy (Kivy, 2016) is an external library for Python that supports Graphical User Interface programming. It provides a simple way to build applications with resemblance to the way Java handles its Graphical User Interface generation.

The application is compiled in a single float layout that contains the logo, a text input box, and a submit button. Statements that are inputted are sent to the server for an appropriate query to retrieve relevant information for comparison with the input string itself. All the computed values that passed the 0.75 similarity threshold will move on to feed data into the computational model. Results are to be displayed in a popup to prompt the user of the input statement's validity and supporting information for further readings (see Fig. 5).

Another external plugin to build the Android package was used. Buildozer compiles a set of Python files and wraps it to create an APK using the standard NDK and SDK procedures. The APK produced is then installed into a smartphone to work independently as a client communicating with the server.



Kylie Jenner has her own show

100.0% Verified

Kylie Jenner has her own show

Kylie Jenner gets her own spin-off show

http://edition.cnn.com/2017/04/10/entertainment/kylie-jenner-spinoff-show-life-of-kylie/index.html

Fig. 5 User Interface of FactU

Program Flow

The application waits for an input IP Address of the server. The application then waits for an input statement or a URL from the user and sends it to the server. Upon receiving it on the server side, the string is preprocessed and is classified whether it is a URL or a plain statement. If the input is a URL meaning it has a net location and a scheme, a spider is deployed in the specific address and the title tag is extracted and parsed to be forwarded as a normal statement.

A function will then extract its subject and the information from the database that contains the same subject is retrieved and is passed on to the SSA module that compares the statement from each of the retrieved information from the database for a score that will be later on used in the HMM module.

The HMM module uses the similarity computed from the previous module to classify to whether the input is verified or satiric. This result is then rendered in the front end app that will display the percentage computed as well as a short list of statements retrieved from the database with high similarity and provides a link for further readings.

RESULTS AND DISCUSSION

15 pairs of sentences are inputted into 3 different approaches each having its own pros and cons. Also, a survey was conducted to further assess these approaches and come up with a decision whether which approach is most suited for classifying news headlines before they are fed into the Hidden Markov Model for final computation of the result (see Appendix A).

NV space approach produces close results when it comes to statements that are predicted by the survey to be significantly the same. However, if the statements contain a related set of keywords but have a different meaning, the approach still gives a high similarity score different from the ones projected from the survey.

The second approach, Semantic and Word order Similarity Approach, was found to be inclined with the expected output based on the survey but, this approach generally gives a relatively low score since effectiveness is limited to short sentences.

The triplet extraction approach outperformed the two other approaches which almost matched the expectation from the survey but requires both of the statements to have a subject, a verb, and an object to perform effectively.

	Triplet Extraction	NV-Space	Without IC	With IC	Expected
1	0.781	0.827	0.682	0.596	0.633
2	0.852	0.995	0.614	0.553	0.747
3	0.744	0.706	0.572	0.525	0.633

4	0.892	0.880	0.626	0.731	0.697
5	0.889	0.868	0.351	0.512	0.810
6	0.807	0.836	0.679	0.831	0.793
7	0.212	0.707	0.317	0.417	0.200
8	0.703	0.686	0.673	0.656	0.413
9	0.510	0.789	0.336	0.499	0.550
10	0.301	0.801	0.675	0.767	0.107
11	0.534	0.864	0.694	0.695	0.323
12	0.048	0.770	0.357	0.541	0.647
13	Error	0.914	0.670	0.774	0.523
14	0.639	0.938	0.673	0.755	0.693
15	0.314	0.951	0.408	0.519	0.727

Fig. 7 COMPARISON OF 4 DIFFERENT SSA ALGORITHMS AND EXPECTED RESULT FROM SURVEY

45 sentences including URLs (see Appendix B) were also tested as input for the program to test its effectivity.

Data gathered from testing resulted in an 77.78% success rate (see Appendix C). Some input resulted in 'Insufficient Evidence' as an output because no entry from the database was able to pass the 75% similarity threshold with the given input, thus, no sense to forward it to the HMM module. On the other hand, some results were erroneous due to having a high similarity percentage with its corresponding database entry but also having a lot of other entries with significant similarities from the other classification making it erroneous. One of the test cases resulted into 'Cannot access link' because for some reason the specific website does not let spiders crawl its documents to protect its information.

These results show the limitations of every Natural Language Processing approach which has been under research for a long period of time. There are various techniques developed however each of them has their own pros and cons.

CONCLUSION AND FUTURE WORK

The field of building Python code for an Android Operating System is still in its infancy. However, it is good enough to develop applications with simple interfaces. On the other hand, Web Crawling has its advanced uses and a wide array of techniques to play with to customize its functionality, although, there is still no definite way of exhausting all relevant web pages to maximize its purpose. In this case, even though there has already been around 5,000 web crawled news articles, some input statements still seem to have insufficient evidence because information computed are only based on news websites, meaning if no article has been written regarding a specific topic, the application will have no means of classifying the statement.

Out of the three approaches to calculate for a score to classify news statements, the Triplet-extraction approach, was found to produce the most realistic results when compared to the results received from the survey and most appropriate values to be used for the HMM.

Although Hidden Markov Model generated a realistic result, other Artificial Intelligence computational models may be used to garner a more definite result. The results discussed also suggest the need for a bigger database with more information to calculate a better result. Also, Natural Language Processing is an open research, as improvement in the way comparison of statements are scored will greatly improve the output of the program.

REFERENCES

Achananuparp, X. H. and Xiajiong, S. The evaluation of sentence similarity measures. Master's thesis.

Agre G. H. and Mahajan N. V. Keyword focused web crawler. 2015.

Bort J. It took only 36 hours for these students to solve facebooks fake-news problem. 2016.

Ghahramani Z. An introduction to hidden markov models and bayesian networks. International Journal of Pattern Recognition and Artificial Intelligence, 2001.

Goel, A. Hackprincetonf16. 2016.

Kashyap, L. Han and Finin T. Robust semantic text similarity using lsa, machine learning, and linguistic resources.

Kivy.org, Python-for-android. 2016.

Kumar R. and Agrawal C. Survey of web crawling algorithms. Advances in Vision Computing: An International Journal (AVC), vol. 1, Sept. 2014.

Lee M. A grammar-based semantic similarity algorithm for natural language sentences. vol. 2014, 2014.

Lee M. and Zhang J. Sentence similarity computation based on pos and semantic nets. Fifth International Joint Conference on INC, IMS and IDC, 2009.

Li Y. and Bandar Z. A method for measuring sentence similarity and its application to conversational agents.

Manku G. S. and Sarma A. D. Detecting near-duplicates for web crawling. Track: Data Mining, 2007

Olston C. and Najork M. Web crawling. Information Retrieval, 2010.

Peisu X. and Qinzhen H. A framework of deep web crawler. 2008.

Ramage D. Hidden markov models fundamentals. Dec. 2013.

Tulchak L. V. and Marchuk A. O. History of python.

Yuntong L. A sentence semantic similarity calculating method based on segmented semantic comparison. vol. 48, 2013.

APPENDIX A

1. US seeks to cool tensions with EU over Gilbraltar.

   US wants to make peace with EU.

2. Ex-Trump adviser Carter Page met with russian intel operative in 2013.

   Carter Page had an appointment with a russian operative.

3. US working with china over North Korea

   US teams up with China to fend off North Korea

4. Russia could soon control a US oil company

   An oil company in US will soon be owned by Russia

5. Passenger wants to file a case

   Passenger plans legal action

6. Stem cells offer hope for autism

   Stem cells may be the solution for autism

7. Yahoo faces chinese dissidents' lawsuit

   China prefers Yahoo over Gmail

8. Facebook targets 30,000 accounts in crackdown on fake news in France

   Facebook showed fake news in France

9. Zebra spotted roaming Riverside County Neighborhood

   Zebra seen strolling around the park

10. Turkey blocks wikipedia for not removing content

    A man eating Turkey blocks wikipedia for content

11. US attorney general vows crackdown on gang violence

    Filipino attorney general vows to stop gang violence

12. When it comes to Syria, the ball's in Trump's court

    Trump controls matters regarding Syria

13. Wanted: sociable hermit for Austrian cliffside retreat

    Austrian cliffside retreat demands sociable introverts

14. Syria chemical attack: Russia challenges Trump

    Trump is challenged by Russia amidst Syria chemical attack

15. Philippines: Duterte welcomes Chinese navy ship visit to Davao

    Navy ship from China receive warm welcome from President Duterte

APPENDIX B

1. US wants to make peace with EU
2. Carter Page had an appointment with a russian operative
3. US teams up with China to fend off North Korea
4. An oil company in US will soon be owned by Russia
5. Passenger wants to file a case
6. Stem cells may be the solution for autism
7. Trump is challenged by Russia amidst Syria chemical attack
8. Kylie Jenner has her own show
9. Zebra seen strolling around the Riverside County
10. Facebook targets many accounts to help keep track of fake news
11. Attorney general from US vows to handle gang violence
12. President Rodrigo Duterte warmly welcomes navy ship from China
13. Wikipedia was blocked by Turkey for not pulling out information
14. Ex-Trump Adviser Met With A Russian Spy
15. Arrest of foreigners increase under Trump regime
16. http://www.complex.com/pop-culture/2017/04/kylie-jenner-getting-her-own-show-life-of-kylie
17. https://news.vice.com/story/russia-oil-rosneft-infrastructure
18. https://www.cnet.com/au/news/facebook-targets-coordinated-campaigns-spreading-fake-news/
19. http://nypost.com/2017/04/06/russia-challenges-trump-after-syria-chemical-attack/
20. http://nypost.com/2017/04/14/solar-powered-device-turns-air-into-drinkable-water/
21. Britney Spears to be taken care by African child

22. www.slate.com/articles/news-and-politics/low-concept/2007/10/save-the-celebrity-children.html
23. Glenn Beck calls for help after eating halal pizza
24. http://www.visajourney.com/forums/topic/413120-glenn-beck-calls-911-after-accidntally-eating-halal-pizza/
25. Palin licks Frozen Flagpole
26. https://theintellectualist.co/sarah-palin-licks-frozen-flagpole-in-iowa-gets-stuck/
27. Landfill was named 'Obama' in North Dakota
28. http://www.a-cnn.com/index.php/articles/item/1322-north-dakota-names-Landfill-after-obama
29. Barry Manilow is a homosexual
30. https://www.theguardian.com/music/2017/apr/05/barry-manilow-reveals-he-is-gay
31. Safest place during tornado is in my arms
32. brain scan can show your dreams
33. http://www.wired.co.uk/article/neuroscience-of-dreaming-consciousness
34. North Korea kills nuclear scientist
35. https://www.democraticunderground.com/1016184083
36. Trump shows Son how to hunt in zoo
37. http://www.scoopnest.com/user/TheOnion/854742206376181761
38. Scorpion stings passenger while in flight
39. UP bestows honorary degree on Aquino
40. Sky got the contract for Easter Week TV
41. Duterte is a murderer
42. Duterte is dead

43.Philippines bans pork

44.Dinosaur found in Palawan

45.Oregano can kill cancer

APPENDIX C

	Computed Result	Expected Result
1	87.08 Verified	Verified
2	81.03 Verified	Verified
3	78.76 Verified	Verified
4	83.20 Verified	Verified
5	81.86 Verified	Verified
6	80.66 Verified	Verified
7	86.82 Verified	Verified
8	100.0 Verified	Verified
9	Insufficient Evidence	Verified
10	90.57 Verified	Verified
11	76.48 Verified	Verified
12	Insufficient Evidence	Verified
13	82.50 Verified	Verified
14	85.00 Verified	Verified
15	90.97 Verified	Verified
16	77.84 Verified	Verified
17	86.90 Verified	Verified
18	77.84 Verified	Verified
19	86.90 Verified	Verified
20	94.20 Verified	Verified
21	82.64 Satiric	Satiric
22	76.67 Satiric	Satiric
23	95.10 Satiric	Satiric

24	100.0 Satiric	Satiric
25	85.00 Satiric	Satiric
26	100.0 Satiric	Satiric
27	75.00 Satiric	Satiric
28	100.0 Satiric	Satiric
29	81.91 Satiric	Satiric
30	100.0 Satiric	Satiric
31	81.91 Satiric	Satiric
32	75.49 Verified	Satiric
33	78.81 Satiric	Satiric
34	88.39 Verified	Satiric
35	80.92 Verified	Satiric
36	88.21 Verified	Satiric
37	Cannot Access Link	Satiric
38	77.82 Satiric	Satiric
39	92.50 Satiric	Satiric
40	89.20 Satiric	Satiric
41	85.93 Verified	Insufficient Evidence
42	Insufficient Evidence	Insufficient Evidence
43	83.89 Verified	Insufficient Evidence
44	Insufficient Evidence	Insufficient Evidence
45	Insufficient Evidence	Insufficient Evidence

Table 2 A comparison of computer scores and expected output