**Natural Language Processing of Clinical Data**

**Giheon Koh, University of Central Arkansas**

**Conway, Arkansas**

**Abstract**

**For the fast adoption of unstructured Electronic Medical Records (EMRs), it is necessary to extract accurate information from EMRs to support automated systems at the point of care and to enable secondary use of EMRs for clinical and translational research. For this research, I designed the Natural Language Processing protocol model by applying machine learning algorithms and I endeavored to figure out the minimum test error rate possible while testing EMR samples in the protocol. While implementing, I used the Python programming language with the Natural Language Tool Kit (NLTK) package as my platform. For the experiment, I used 20 de-identified pathology report samples. For the workflow, I searched and extracted cardinal quantity values from the samples, then extracted the strings for the final diagnosis. When extracting the diagnosis information, I used the Support Vector Machine (SVM) algorithm. While working in the SVM model, I calculated the test error to verify how significant the extracted information is. For the entire protocol, I designed it based on a decision tree algorithm to extract necessary information from the unstructured EMRs. As the result, I extracted the patient’s name, medical record number, gender, age, affected body part, procedure, diagnosis description based on 10% error rate, grade, date and note ID information from each sample, and stored them in a database.**

**Introduction**

The healthcare system has grown rapidly over the last century. This growth required that the relevant healthcare data were adequately and appropriately stored alongside the evolution of healthcare technology. As a tool adapted to the technology, the Electronic Medical Record (EMR) system has provided clinicians a host of benefits in terms of understanding patients by tracking their histories from many healthcare providers. Through the EMRs, clinicians were able to efficiently identify general information such as treatments and medical histories of patients. However, with the rapid adoption of unstructured or free text-formatted EMRs, which contain more detailed information about patients than do the structured, certain concerns were brought to light. First, because of the staggering number of medical records, clinicians had to spend a great deal of time reading, classifying and summarizing information for each patient. Second, it was imperative that valid and accurate information be extracted from EMRs to support clinicians’ decisions at the point of care. Third, it was desirable to enable secondary use of EMRs for clinical and translational research. The Natural Language Processing tool was developed to address these concerns and facilitate information extraction from such unstructured free texts.

Laurie Miles, the head of analytics for big data specialists at SAS [1], stated, “About 75% of data is unstructured, coming from sources such as text, voice, and video."Due to the massive amount of data hidden in unstructured texts, it became essential to develop tools to extract information. Natural Language Processing (NLP) presents itself as the solution. According to the definition of Natural Language Processing by SAS institute Inc. [2], NLP is “a branch of artificial intelligence that helps computers understand, interpret, and manipulate human language.” Thus, NLP helps clinicians interpret free text or human language and make it analyzable.

A brief overview of the history of Natural Language processing is in order. According to Nadkarni et al [3], NLP started in the 1950s as an intersectional tool of computer and linguistics. In 1956, Noam Chomsky published a book, *Syntactic Structures*, which introduced such revolutionary linguistic concepts as that positing that a computer can understand human language. Since then, many programming languages have been proposed and developed. After fourteen years, Natural Language Processing research was linked with statistics for the higher and more detail-oriented demand of linguistics analysis. There were many proposals for the neural language model in the 21st century as well. A prime example is Apple’s Siri, the first successful NLP and artificial intelligence assistant. NLP research today is still an ongoing project in minimization of test errors and extraction of information as accurate as possible from unstructured sources.

According to Kreimeyer et al. [4], the utilization of Natural Language Processing for text mining offers many advantages. First, it will help to reduce time for manual expert review. For clinicians, processing EMRs for patients is a highly time-consuming task. However, when done with an automated processing system that converts unstructured data to structured form, much time will be saved in expert review and there will also be more flexibility in secondary use of such data for large-scale automated processing. Utilizing NLP is also advantageous for gaining more knowledge about patients. Sometimes, it may happen that a clinician misses important information from a free-text medical report. To prevent this, NLP will extract and organize all necessary information and store it into a database.

Despite these advantages, NLP still has some challenges. Because of a free text’s poor structure, abundant shorthand, and domain-specific vocabularies, it is quite difficult to figure out zero test error rate, meaning that some cases miss capturing important terms or otherwise capture unimportant terms. For this research, I designed an NLP protocol model by applying machine learning approaches with a focus on determining the minimum test error rate possible when testing EMR samples in the protocol.

**Important Concepts**

Certain basic NLP concepts should be reviewed in order to gain a better understanding of the Natural Language Processing protocol. There are many concepts in NLP but studying a few important concepts with examples may shed light on the entire NLP protocol. The listed concepts are arranged based on the procedure of building up the protocol: Corpus, Tokenization, Stop Words, Normalization, Stemming, Lemmatization, and Part-of-Speech tagging.

The first concept is the corpus. A corpus is a body of written or spoken material upon which a linguistic analysis is based, meaning that statistics is accumulated on natural language text. In practice, the medical record sample is generally referred to as the corpus.

The second concept is tokenization. The idea of tokenization is to process word recognition by splitting strings into smaller pieces, called tokens. As shown in *Figure 1*, a string is split into tokens.

A screenshot of a cell phone

Description automatically generated

Figure 1. Tokenization

The third concept is the stop words. The stop words generally refer to the most common words in a language, which contribute little to the overall meaning and therefore should be filtered out before further processing of text. Main examples of stop words include ‘the’, ‘a/ an’, ‘of’, ‘and’, ‘or’, ‘am/ is/ are’, and others. Thus, for the general process as shown in *Figure 2*, the stop words are deleted from the tokenized sentence.

A screenshot of a cell phone

Description automatically generated

Figure 2. Filtering against Stop words

The fourth concept is normalization. Before further processing, texts need to be normalized, meaning that the testing samples should be set on equal footing. The process of normalization mainly refers to the tasks needed to do this, such as converting all texts to the same case (upper or lower), removing all punctuation, expanding contractions, and converting numbers to their word equivalents. For example, *Figure 3* shows that uppercase letters are converted to their lowercase forms and all punctuation is removed.

**A close up of a logo

Description automatically generated**

Figure 3. Normalization

The stemming is the process of eliminating affixes such as suffixes, prefixes, infixes, and circumfixes from a word to obtain a word stem. The stemming is necessary in letting a computer recognize that some words having different word forms are fundamentally the same based on the words’ stems. For example, there are two words: ran and running. For a computer, they are different because of their different spellings and lengths. However, when looking over their word stems, they are fundamentally the same. Thus, through the stemming process, a computer can recognize that those words mean the same thing. In practice, *Figure 4* shows an example. The words, ‘capital,’ ‘populous,’ ‘city,’ and ‘united’ are converted to their stem words.

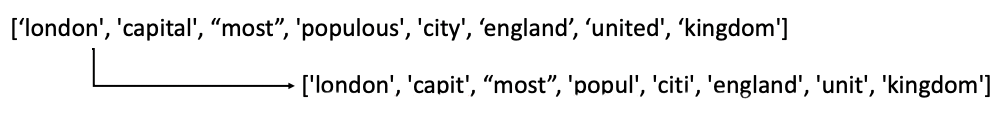
****

Figure 4. Stemming

Lemmatization is similar to the stemming process in that the purposes of stemming and lemmatization are the same. However, the distinguishing factor is that lemmatization captures the canonical forms based on a word’s lemma. In the linguistic definition, a lemma is *“the base form under which the word is entered in a dictionary.”* For an example of lemmatization, a word, ‘better,’ will be processed to ‘good’ after lemmatizing. In the practice shown in *Figure 5*, the word ‘most’ is converted to ‘more’ based on its word lemma.

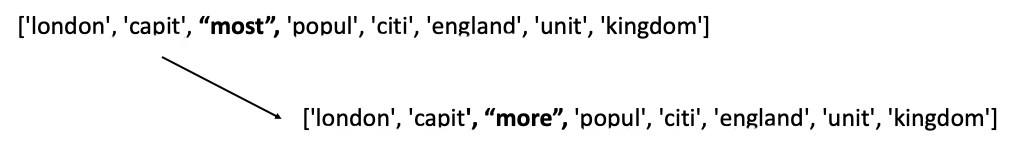


Figure 5. Lemmatization

For the last concept, the Part-Of-Speech (POS) tagging is the process of assigning category tags to the tokens of a sentence based on the general rule of Part of Speech. When conducting the POS tagging to each word, a computer is assisted in recognizing what tokens to extract. In this process, what a computer mainly takes care of are the nouns, verbs, and adjectives. In *Figure 6*, ‘NN’ means the common noun, and ‘ADV’ means adverb. Additionally, there are many kinds of nouns, verbs, and adjectives under the main branches of the Part of Speech rule.

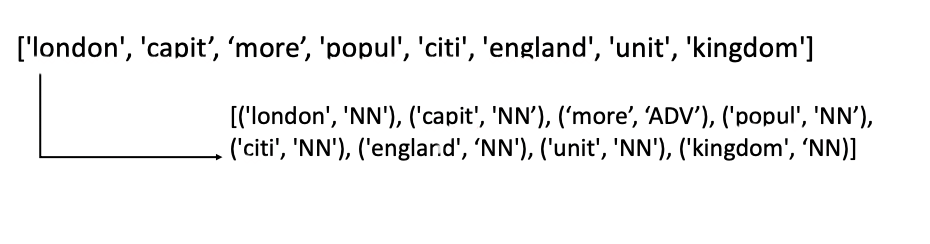


Figure 6. Part-Of-Speech (POS) Tagging

**Technology Applied**

For this research project, I used the Python programming language because the Natural Language Processing (NLP) requires Machine-Learning (ML) algorithms for automated processing of large-scale data, and Python provides both these NLP and ML tools in one platform. For the NLP tool, I used the Natural Language Tool Kit (NLTK) package because of its utility and popularity.

**Samples Description**

For this experiment, I used 20 de-identified pathology reports. This research was not able to use real-world pathology reports because they contain real patients’ confidential information. As shown in *Figure 7*, the samples are quite unstructured and have different arrangements of information. This is because the structures of such pathology reports depend on patients’ diagnoses and the providers that entered the text.

**A close up of a newspaper

Description automatically generated**

Figure 7. Examples of deidentified pathology reports

**Workflow**

The workflow is divided into four stages: Preprocessing, Information extraction based on Part-Of-Speech (POS) tags, Information extraction from diagnosis description, and Exporting the mined data.

According to Assale et al. [5], the preprocessing stage includes data cleaning, data integration, data reduction, and data transformation. The stage has two major aims: to make data cleaner in terms of noise, inconsistency and incompleteness, and to improve the speed and accuracy of data mining, dealing with heterogeneous data and its redundancy. In practice, importing the stemming, lemmatization standards, and the list of stop words is the first step. These standards and list are provided by the Natural Language Tool Kit (NLTK) package in Python. Then, importing the Unified Medical Language System (UMLS) is the second step. The UMLS is a compendium of many controlled vocabularies in the biomedical sciences and used to train testing samples to extract only important terminologies in a diagnosis description. After these systems are imported, building up the algorithm to test sentence-by-sentence in a testing sample and word-by-word in a processing sentence is the next step. The mechanism of this algorithm is that for a pointer in looping for sentences iterating, the pointed sentence is tokenized, and another pointer inside the sentence is generated. For iterating words in the tokenized sentence, the pointed word is tested against stop words to determine whether it must be filtered out. Then the filtered words are lemmatized. After the process, the Part-Of-Speech tags are assigned to each word.

In the information extraction based on the POS tags stage, the algorithm is designed to search for cardinal quantity values, which are string-based numeric values and categorized as nouns. The reason the algorithm searches for the cardinal quantity values is that the numeric values in medical reports are typically more meaningful than most of the words. On average, the age, medical record number, and procedure date and time are extracted when extracting the cardinal quantity in a sample. After that, the algorithm searches for gender information among the nouns. For this, I set a list of words indicating gender, meaning the list of stem words such as man, woman, gentleman, lady, and others. Then, based on those words, the algorithm searches for the gender information. *Figure 8* shows the table for extracted information based on the POS tags.

A screenshot of a cell phone

Description automatically generated

Figure 8. Table for Extracted Information based on the POS tags

For the information extraction from the diagnosis description stage, the first step is to extract the strings for the final diagnosis from a sample. Throughout all samples, there is a common indicator for the final diagnosis section. For example, most of the final diagnosis sections are in the following form:

… FINAL DIAGNOSIS : A. Cervix, biopsy: - At least high grade squamous intraepithelial lesion (CIN III) in a background of extensive necrosis, see comment. Comment : The specimen is predominately necrotic; however, there are small superficial fragments of severely dysplastic epithelium present. These findings are consistent with squamous cell carcinoma (unsampled) ...

From the sample, there is the section indicator at the head – here, ‘FINAL DIAGNOSIS’. Based on the indicator, the algorithm extracts the strings for the final diagnosis section. From the extracted strings, the next step is to compart the affected body part, clinical procedure information, and diagnosis descriptions for each alphabetical index whose numbers depend on the numbers of diagnoses. When finding the minimal common pattern in the final diagnosis for each index, the next string after every alphabetical index is the string for the affected body part, the following string is the clinical procedure, and the strings behind the hyphen are the diagnosis description for the body part in the same index.

As the affected body part and clinical procedure information are extracted, the strings for the diagnosis description should be filtered for the important terms. For this, the Support-Vector Machine (SVM) algorithm is applied. SVM is one of the machine learning approaches that analyzes data used for classification or grouping. The method of classification is the SVM algorithm’s creation of a linear regression model-based optimal compartment called a hyperplane between the data points. The algorithm then generates the support vectors at the closest data points on each comparted side. The test error is calculated as proportional to the margin between the support vectors and hyperplane, meaning that more margin results in more significance. For this research, SVM is used to calculate the test error of classification. Since SVM is a supervised learning machine learning algorithm, a training dataset needs to be defined before conduction of SVM. Hence, UMLS, which is imported in the preprocessing stage, is put as the training dataset. When the training dataset is defined, each word is tested for the word’s matching rate with the words in UMLS. The matching rate is simply calculated the proportion of matching words. For example, supposing that the testing value is ‘squamous cell carcinoma’ and the value in UMLS is ‘buccal squamous cell carcinoma’, the matching rate of testing value over the value in UMLS is 0.75.

A close up of a piece of paper

Description automatically generated

Figure 9.

1. Matching rate table for testing words and UMLS-lexicons, 2. Transformed table for the input of SVM model, 3. The conceptual model of SVM

In practice, filtering the result table for the matching rate greater than or equal to 0.75 yields *Figure 9-1*. To fit this table as the input of the SVM model, the table should be transformed into a contingency table *(Figure 9-2)*. When the table is visualized, it presents as *Figure 9-3*. In *Figure 10,* the hyperplane is created based on the linear regression model and the support vectors are generated at the closest data points on each side. In the final output, the terms upon the upper support vectors are chosen to be tested for their significances.

A close up of a map

Description automatically generated

Figure 10. The practical model of SVM

When the SVM model is set, the algorithm calculates a test error of the SVM model. *Figure 11* is the recommended formula from the research of Gaonkar and Davatzikos (2013) [6] for the approximation of permutation testing for SVMs. In the formula, *E*[*P* (*Error*)] is a measure of the generalization or test error of the SVM, E[Number of support vectors] is the mean number of generated support vectors for every permutation, and the number of training samples is the number of permuting samples. Therefore, for the example in *Figure 10*, there are on average only two linear support vectors when permuting values because those support vectors are based on the linear regression model of data points and the lower support vector is always laid at zero on z -axis. The number of training samples is the factorial of the testing samples, which is 20, so we get, 20! = 2.432902008 E+18. Therefore, *E*[*P* (*Error*)] ≤ 1/20! ≈ 0. For the result of the example, the test error of the SVM model is less than or equal to 0. For all tests, the terms of the diagnosis description in the final output are chosen based on the models which have test error less than 0.25 in consideration of important but wrong-spelled words by sample providers.

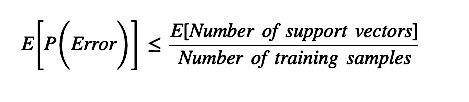


Figure 11. Formula for test error of SVM with permutation

For the last stage, the algorithm reanalyzes and extracts the information for the medical note ID, the medical record number of a patient, and age information from the table as extracted information based on the POS tags. Then, it finally merges all the harvested information in one data frame and exports it to a Microsoft Excel file.

The entire protocol is designed in accordance with the decision tree in *Figure 12.* Every word after filtering out against stop words and normalizing is tested for its types of classification based on the unit defined, the set of indicators, and general definitions in medical records. When a word reaches a node, and if the node is one of the underlined, the word is either stored in the final result or skipped.

A close up of a map

Description automatically generated

Figure 12. Decision Tree Model

**Final Result**

After proceeding through the workflow, we generate the outcome of processing 20 de-identified pathology reports in the protocol. For each column, it refers to the patient’s name, medical record number, age, affected body part, procedure, diagnosis description, grade of diagnosis, procedure date, and medical record ID.

**A close up of text on a white background

Description automatically generated**

Figure 13. Final Output

**Challenges**

While building up the Natural Language Processing protocol, it was quite challenging to deal with its unstructured format, abundance of shorthand, incorrect grammar and spelling, and misuse of hyphens, among other errors. Because the content of the same type of report can be organized differently depending on patients’ individual diagnoses or the provider that entered the text, a high-test error rate often ensues. For future research, methods of building on the advanced protocol dealing with these variances will be studied.

**Conclusion**

Because of the rapid adoption of the unstructured Electronic Medical Record, Natural Language Processing has accordingly been developed to extract important and dependable information from such free texts. For this project, I applied Support Vector Machine to extract accurate and important terms from the texts and decision tree algorithms to automate the processing of large-scale data. As a result, I extracted the patient’s name, MRN, gender, age, affected body part, procedure, diagnosis description, grade, date, and note ID. While doing the research, there were many challenges in dealing with high variance in the free texts. With more research into natural language processing cases, approaches, and protocols, further development of the advanced NLP protocol will continue in future.

**Acknowledgments**

This project was supported by the Arkansas INBRE program along with the National Institute of General Medical Sciences -NIGMS (P20 GM103429) - from the National Institutes of Health. I would also like to acknowledge Dr. Ahmad Baghal, Dr. Shaymaa Al-Shukri, and Annu Kumari in the Department of Biomedical Informatics at UAMS for their valuable support and guidance in the project.

**References**

[1] Wall, M. (2014, March 04). Big Data: Are you ready for blast-off? Retrieved June 18, 2019, from https://www.bbc.com/news/business-26383058

[2] What is Natural Language Processing? Retrieved July 9, 2019, from https://www.sas.com/en\_us/insights/analytics/what-is-natural-language-processing-nlp.html

[3] Nadkarni PM, Ohno-Machado L, Chapman WW. Natural language processing: an introduction. *J Am Med Inform Assoc*. 2011;18(5):544–551. doi:10.1136/amiajnl-2011-000464

[4] Kreimeyer, K., Foster, M., Pandey, A., Arya, N., Halford, G., Jones, S. F., . . . Botsis, T. (17 july 2017). Natural Language Processing Systems for Capturing and Standardizing Unstructured Clinical Information. *Journal of Biomedical Informatics,*1-16. doi:10.1016

[5] Assale, M., Dui, L. G., Cina, A., Seveso, A., & Cabitza, F. (17 april 2019). The Revival of the Notes Field: Leveraging the Unstructured Content in Electronic Health Records. *Frontiers in Medicine,*1-23. doi:10.3389

[6] Gaonkar B, Davatzikos C. Deriving statistical significance maps for SVM based image classification and group comparisons. *Med Image Comput Comput Assist Interv*. 2012;15(Pt 1):723–730.