EXTRACTING JAMAICAN GEOGRAPHIC LOCATIONS USING THE NLTK AND PATTERN TOOLKIT FOR PYTHON FROM NEWS ARTICLES

Extracting Jamaican geographic locations using the NLTK and Pattern Toolkit for Python from news articles

# Jean-Mark Wright

# Abstract

Natural Language Processing (NLP) has long been used to extract information from large bodies of text. Natural Language Processing is often desirable to intelligently parse large volumes of data where the manual alternative may be infeasible. Using Named Entity Recognition, NLP algorithms can be created to extract the mentions of geographic locations of different types from current and archived news articles. This information can be used to auto plot these locations on the map and provide datasets for deriving how hot a particular location is.

# Title of Paper

Extracting Jamaican geographic locations using the NLTK Toolkit for Python from news articles

# Background

Natural Language Processing (NLP) has long been used to extract information from large bodies of text. Natural Language Processing is often desirable to eliminate the need for manual perusing of such sources. The ability to parse human written information produces many economic and practical gains. Even more compelling is the ability to understand the written text. When algorithms are able to understand written text, they can be used to process large volumes of data and make derivations based on the data. This approach has many merits when compared to a manual approach. With the use of training sets, computers can be taught and can arguably be more consistent, focused and unbiased than their human counterparts. Sentiment extraction and analysis; the process of extracting meaning from natural language, is a fruitful area of research within NLP that has also been heavily researched. Many studies and approaches have been presented to extract polarities from written text. NLP has been used to perform document level sentiment summarization and also producing more granular derivations based on sentences or phrases written in natural language. Such algorithms have been used to power commercial grade systems with very high levels of accuracy and precision ([Jeonghee Yi† [1](#_ENREF_1)]). Information that is derived can be used to support intelligent decision making, influence expert systems, and even trigger early warning systems.

Natural Language Processing has been and continues to show great potential for bridging the gap where there is no established format of communication. Traditionally, parsers and their grammars have dictated context and are constructed to avoid ambiguity. However, using NLP, the computer can parse and understand written text across several languages despite language idiosyncrasies and ambiguity. In a sense, ambiguity can be embraced as a normal occurrence that is expected and does not hinder the capturing of accurate and precise conclusions. Also, using NLP, unformatted natural language text can be even be standardised. [CAROL FRIEDMAN [2](#_ENREF_2)] discusses approaches used to automatically encode clinical documents, mapping them to coding used by the Unified medical Language System (UMLS); a well-known and widely used set of codes used in the medical domain.

NLP despite its obvious usefulness represents a domain that is riddled with many challenges. One major challenge presented is the threat to accuracy and precision caused by the absence of context. Ideally, the context of a body of text needs to be derived in order for any meaningful information derivation. This arguably is also necessary for less ambitious approaches that aim just to search natural language for occurrences of a word or phrase. Under different contexts, a single word may have several meanings, requiring that algorithms address ambiguity in documents. To understand exactly what is being described or referenced, a context is often necessary. Obtaining a proper context however still remains an open and challenging problem. Many researchers have avoided this route altogether and instead relied on purely statistical methods instead of attempting to derive a proper context from the text. To understand even a simple sentence often requires knowledge that may or may not be present in the sentence. It follows that to understand a larger article; much of the context required for the article may not be in the article. Also, even when sufficient information exists in an article, it may not be contiguous, connected or structured in a way that makes for simple construction of context using algorithmic approaches. Different methodologies have been employed in research to either directly tackle this problem or avoid dependence on context by using approaches that rely on training sets or other statistical methods. A direct approach to this problem was proposed in [Jeonghee Yi† [1](#_ENREF_1)]. They used a training set and dictionary of sentiments to aid in the understanding of the context. One indirect approach is that of IBM’s Nominator ([Ravin [3](#_ENREF_3)]). Nominator used aggregation of candidate terms across several documents to replace the need for developing a context. Both approaches saw high levels of precision and accuracy.

Named Entity Recognition, is a particular branch of NLP that is also tasked with the challenge of disambiguating information to extract reliable information. Named Entity Recognition is defined as the process of identifying specific names mentioned in a body of text. In general, the three categories recognized are names of persons, organisations and geographic locations. While George Bush and Kofi Annan may immediately be identified as familiar persons in history, it was also noted that Bush is a city of Louisiana and Annan is the name of a small city in Scotland. This example gives a clear picture of the kind of ambiguity that thwarts the accuracy and precision. Some semblance of context or alternative approach is needed to understand exactly what is being referenced. Clashes can also occur between the names of places, persons or organisations and other common words in spoken text. One such prime example; Welcome, is the name of a town in St. James and She is the name of a city in India.

Named Entity Recognition however is an area also ripe with potential. There is a direct application of NER for geocoding; news stories, articles and other user produced text can be geo-tagged. With NER algorithms, very large archives and current issues of news item can be parsed relatively quickly. The information garnered from the process can be used to track mentions of locations as time progresses. Despite the many challenges Natural Language Processing has seen many successes; much of the work reviewed has achieved remarkable rates of precision and accuracy. The measure of accuracy and precision are often described using the F-measure. This measure was used by researchers when measuring their results. The results of the research thus far in NLP present a very strong case for its tangible relevance and usefulness as a technology.

# Research Aims

In general, the aim of this research is to combine techniques used in previous works in order to extract the names of Jamaican geographic features from news articles. We will assume that an article may contain multiple references to geographic locations and will concentrate only on those references within Jamaica. These references will be extracted and tagged to the article. In order to accomplish this, the following steps need to be accomplished:

* Build a corpus that contains confirmed references to geographic locations in Jamaica. This corpus will be built with articles from the Jamaica Gleaner’s website. Requisite tools will be built to extract information from these sources in text. This information will be used to build a training set from which derivations can be made and insights gained to build and train a system to recognize references to Jamaican geographic locations in news articles.
* Build a gazetteer for places in Jamaica. A gazetteer is a spatial dictionary of geographic locations. A basic gazetteer will store location names and a corresponding spatial geometry for that location. This geometry can be a polygon (representing the boundary of the location), centroid (point representing the centre of a boundary) or point representing the latitude and longitude position of the location. No gazetteers are currently available that contains a comprehensive set of places in Jamaica. For the purpose of this research, a minimal gazetteer will be built consisting of parish names and community names. The community dataset consists of the name, parish name and a centroid (coordinate representing the latitude and longitude position of the centre of the community). The parish attribute on the community is used to determine what parish the community is in. The parish dataset has a name and centroid. All locations in the gazetteer will be given a class and a weighting. The class is used to indicate the administrative boundary level, and the weighting is used as an input to the algorithm used to extract location references in an article. Parishes in the gazetteer will be given a class of one (1) and the highest weighting. A community will have the highest class but the lowest weighting. Both parish and community datasets were obtained from the Mona GeoInformatics Institute.
* Develop an algorithm that can perform unsupervised recognition of geographic locations in Jamaica (given they exist in the gazetteer), achieving F-measure of 0.80 or higher. This solution will be developed in the Python programming language using Python’s Natural Language Toolkit (NLTK) and the Pattern toolkit.

# Literature Review

## The Perseus Digital Library

[David A. Smith [4](#_ENREF_4)] set out to disambiguate place names in Historical Digital Library. In their research they described the Perseus digital library and evaluated its performance. In their research they highlighted many of the practical reasons for wanting to utilize Natural Language techniques. In particular they highlighted the infeasibility of the manual tagging of vast libraries of geographical data. Automated geocoding is a much preferred approach. The process of geocoding mainly involves two broad steps: identification and categorization and also disambiguation. Primarily, they noted that the wide range of data directly affects the process of disambiguation. They made used of the gazetteer to disambiguate the results found. It was noted that systems whose task is to match entities such as personal names or company names can easily operate without a list of known names. However, for the process of identifying geographic locations, it is necessary to obtain an index such as a gazetteer that supplies names and location based information for each place.

In their review, they spoke of several systems built. BBN's Nymble has a 0.9 score for F-measure. It however requires a large amount of training data. IBM's nominator came close the Nymble with a score of 0.88. Another system reviewed was the Geo-Referenced Information Processing Systems (GIPS). GIPS identifies geographic locations without performing disambiguation. It relies rather on the process of aggregation of the information over several articles to filter out ambiguity. “Reducing ambiguity” was also highlighted as causing a problem in entity recognition. Initially a document may say "President Bill Clinton" but there after refer to the President as "Clinton" or "Mr. Clinton". This presents a problem for name recognition. They also highlighted the fact the names of persons may often clash with the names of geographic locations. Approaches that ignore timelines also present a challenge on the relevance and accuracy of the information. It was noted that across the different continents, there are several places that have multiple names, and also the percentage of names that clash with places names were high in certain regions. It was found that the percentage of name clashes were as high as 57.1% in North and Central America.

To perform the procedure of disambiguation, internal and external sources are considered. The external sources used were gazetteers, bibliographic information and general linguistic knowledge. Internally, honorifics, the linguistic environment and geographic labels are used to aid in the disambiguation process. A listing of over one million names was constructed for the purpose of disambiguation and name identification. Perseus uses the rule of thumb that candidate proper names are generally capitalized. Perseus uses the existence of honorifics as strong indicators of the presence of a name i.e. the presence of topographic labels e.g. mountain or river were used as a strong indicators of the presence of a geographic feature.

The first step in their procedure for identifying and disambiguating place names was to search for all named entities and tag them name, place or date. Their system next looked up all entries tagged as geographic names in the gazetteer. Uncertain names are also matched. A context for the document was determined by the major influence of the dominating or repeated geographic name(s). A score is also calculated for each result based on the proximity to other locations and relative importance (parishes have a higher importance than a city or town). The authors calculated the centroid of the article based on the different locations mentioned. They then calculated the standard deviation of the distances from the centroid. Points that were found to be two or more standard deviations away from the centroid were discarded then a new centroid calculated. After the centroid is created, a context consisting of the four geographic references before and after a reference is created. Each geographic reference is now given a score based on its proximity to the geographic references in its context, distance from the new centroid and also the relative importance. The candidate geographic reference with the highest score at the end is deemed the winner.

When tested using standard precision and recall means, the system scored very well, getting as high as 0.93 in precision for Greek places and a high of 1.00 in recall for Roman names. The F-measure was highest for Greek names, with a value of 0.96. Scores obtained for London, California and the Upper Midwest were lower. This was attributed to the duplicity of names and places. The author noted that proper names are not generally inflected in English, and hence it was deemed that the use of stemming would do more harm than good for the research. More information could have published about the details of the testing procedures. The authors did explain that their tests were biased toward system recall and that corpuses were used for different languages for the evaluation of the system and that humans were also used to manually work through 20% of the corpus of each language. They also provided analysis for the results obtained.

## Nominator

[Ravin [3](#_ENREF_3)] in their work on Extracting names from Natural Language Text, aimed to supplement the technologies used at IBM. They built a system called Nominator for this purpose. Nominator identifies proper names in text, classifying them as place, name or organization. It was built to deal mostly with data generated from news items.

When names are identified in a body of text, each group is assigned a canonical name. The canonical name is chosen to be the least ambiguous and most permanent name. The canonical name serves for a grouping of names. Hence, many words or phrases may refer to a single name. Nominator uses 1600 tags and other name indicators (e.g. Mr., Mrs.), 700 most common places names and approximately 20,000 first names. Nominator also relies of capitalization, punctuation and on contextualization to adequately classify names. This assumption they based on the conventions normally observed in the texts that are properly edited. Nominator also assumes that most names are referred to first in its most explicit sense then are referred to using shorter forms. An important point made in their research was that names often are phrases that cannot be separated; they must be preserved to keep their meaning.

In their review, very little was mentioned of previous research. They however did briefly mention the StenStep application. StenStep performs tokenization of its input. This process is however minimal.

In order to extract names from a document, Nominator performed the following steps:

* Parse the document for a list of candidate names. The document presented was first searched to generate a list of potential candidate names.
* Split the names into smaller names. After parsing, a second perusal the names are compared to each other. In this phase, some names are discarded and others are edited or adjusted. This second parsing is used to narrow the list.
* Group the names into equivalence classes. The list is then divided into groups of equivalent names and a canonical name is selected for each group. Nominator produces a database of names which are each related to a collection of documents.
* Aggregate the classes across multiple documents. After all the documents have been analyzed and canonical names have been selected for each group, groups with the same canonical name are merged. When the aggregation is done, the stronger canonical name always overrides the weaker. This aggregation is used as an indirect approach to combat ambiguity in the document.

IBM's approach in building Nominator focused less on intensive Natural Language Processing techniques and more on simpler operations based on characters and strings. Nominator also does not perform intensive analysis of contexts. In selecting this approach, the authors were able to gain much more speed in processing but limited Nominator's ability to understand what is being discussed or directly tackle ambiguity. The method used instead to tackle ambiguity was to aggregate the equivalence classes across many documents. By taking this approach, the aggregation helps to remove the noise presented by ambiguities.

Nominator was tested using 88 articles in the Wall Street Journal. Proper names were first manually identified from the articles, and then Nominator was given the text to identify proper names. In testing the Nominator performed very well, achieving a high recall of 97.8% and a precision of 91.8%.

The researchers from IBM presented a solid case that was well tested and very well documented. Even though, the approach negated the use of a direct context, the use of aggregation and equivalence classes proved to be a useful and remarkable approach. This was evidenced in the high scores of recall and precision received in the tests performed.

## Geocoding across multiple languages

[Bruno Pouliquen and Blackler [5](#_ENREF_5)] aimed to pick up references to geographic place names in multiple languages. They aimed not only to state that a name is a place, but also to disambiguate homographic place names. One of the primary approaches taken by the authors here was to only tag a place as such after is has been confirmed that the place is not instead the name of a person. Place name identification is done with the help of a gazetteer. The following sources were used to form the gazetteer:

* Global Discovery 2006 – this document contains over 500,000 place names world-wide. This also contains a class attribute that assigns a weighting to all the places.
* KNAB database – this is an online database contains a comprehensive list of geographic locations across the world.
* European commission internal document – this document stores the names of capitals, inhabitant names, currency and country adjectives used.

After places are identified in the gazetteer, the process of disambiguation is run. The process of disambiguation takes into consideration that a place name may clash with the name of a person. The disambiguation process also considers the importance of the place (nation as opposed to city), the main country or countries that the text is about, the presence or absence of geo-stop words, the words in the surrounding lexical contextand the minimum distance to any other places. After person names have been recognized in the text, those names are not considered as candidates for geographic locations. A weighting was assigned to geographic names in the Global Discovery. Countries, country capitals and main cities were all given a high rating of 80. Province level or small cities were given weightings of 30 and 20 respectively.

Shallow geoparsing is first used to identify the highest classes of geographic names in the document. Then deep geoparsing is used to recognize lower class geographic names only if they exist within one of the names previously identified while performing shallow geoparsing. Therefore cities are only retained in the deep geoparsing process if its country or some geometric place which contains it had previously been identified during shallow processing.

A geo stop word tool was also built to help with the ambiguity caused by words being homonymic with each other. The stop word list uses two approaches. The first analyses a list of first names and tags those that are potential names of geographic locations to be stop words. The second approach uses a corpus to identify those homonymic words that occur above a certain frequency and tags them as stop words.

Use of the lexical context was initially used (e.g. using "city" to indicate the existence of a place name), however the authors decided to forgo this approach due to observed negligible benefits and the recognition that part of speech (POS) tagging and other language dependent rules would be required to obtain a more impacting outcome.

The minimum kilometric distance was also used to disambiguate place names. The distance was measured from non-ambiguous places to ambiguous places. This process helped to better target ambiguous references to geographic names.

The authors built their own test set of articles in multiple languages due to the absence of a proper corpus. 162 newspaper stories were selected that were in different languages. The system built had a high precision score of 91 for English with the lowest score for precision being 68 for French. Italian had the highest score for recall, that of 85. German had a low of 68. The highest and lowest scores of F-measure were recorded for English and French with scores of 84 and 70 respectively.

## Sentiment Analysis

[Jeonghee Yi† [1](#_ENREF_1)] sought to use a more direct approach to parsing natural language. Most of the approaches taken attempt to summarize the polarity of the document instead of finding polarity of individual sentiments. That approach possesses an inherit weakness since such summarized polarities fail to capture individual specifics that would often be useful for users of the information. The association of a sentiment to a topic has also been recognized as a difficult problem. At the time of their research, none of the work reviewed had been ground breaking in that regard. Most of these methods have either depended on a classifier to classify the document or used statistical methods based on terms used in the same context.

Their approach primarily focuses on a topic specific feature term for extraction. Processes of sentiment extraction and subject-sentiment association are performed. The approach uses a sentiment lexicon and a sentiment pattern database. A subject can be either a topic of interest or a feature of the topic. For example; a “camera” may be the topic of interest and “battery” may be a feature of the topic.

The Sentiment Analyzer they designed extracts topic specific features, sentiments of each phrase that contains a sentiment and finally makes relationships between the topic and the sentiment.

A feature term was defined as a term that is either:

* A part of the topic (e.g. the battery of a camera)
* An attribute of the topic (e.g. the cost of the camera)
* An attribute of a known feature of the topic (e.g. the life of the camera’s battery)

Authors used a mixture model and likelihood test methods to extract features from the articles. The likelihood tests outscored the mixture model tests. The authors built their sentiment word collection using data from General Inquirer, Dictionary of Affect of Language and WordNet. An observation was made that feature terms are nouns. Hence, they proceeded to extract noun phrases from document. The researchers properly outlined the research process and the components of the sentiment analysis algorithm.

For testing, the authors used a corpus that was created from different web pages containing reviews about cameras. Half of the cases were predetermined to be favourable. Over 2000 reviews about cameras were used to test the system for commercial grade applications. Even though only one subject was selected for testing, the authors sought to explain failures or present justifications for unexpected results that were achieved. In their results, the system was able to achieve a high precision of 95% with a low recall of 24%.

## Named Entity Recognition using Classifier Combination

[Florian, Ittycheriah [6](#_ENREF_6" \o "Florian, 2003 #14)] in their paper sought to combine separate techniques under different conditions to perform named entity recognition. They used the following classifiers for the combined task of recognition.

* Transformation Based Learning (fnTBL)
* Hidden Markov Model (HMM)
* Robust Risk Minimization Classifier (RRM)
* Maximum Entropy Classifier (MaxEnt)

Each classifier has its unique methodology and set of heuristics for performing recognition of named entities. The author expressed that differences in how the classifiers work make them good candidates for combining. The fnTBL classifier bases its decisions on the most influential or deciding features, while the others based their decisions based on a combination of all the features. The search methods are different; the HMM, RMM and MaxEnt classifiers start by building a model then they apply an algorithm to determine the search path. fnTBL on the other hand, generates model dynamically, starting with frequency data. Finally, fnTBL and RRM return only a single classification for each example while MaxEnt and HMM return probability distributions.

All the classifier algorithms seek to tag words in the input based on their proximity to named entities. Words are tagged as either starting, continuing, ending or having no relation to a named entity. In addition, it was noted that access to a rich feature space is necessary for high performance of the classifier. Each classifier has access to the following features while in context:

* A word and lemma for the current word and up to five (5) words surrounding it.
* The part of speech tags for the current word and surrounding five (5).
* Prefixes and suffixes for the current and surrounding words.
* A flag describing the word, such as capitalized, sentence cased or having digits.
* Information from a gazetteer consisting of over 80 000 proper names, 50 000 cities and 3500 organization names.
* The output of two well-trained named entity classifiers.

The combination of the classifiers resulted in very high F-measure scores. Scores for location and person name recognition scored above 0.9, while miscellaneous and organization scored slightly lower in the range 0.8 – 0.9 for English.

## NER without a gazeteer

The work done here by [Mikheev, Moens [7](#_ENREF_7" \o "Mikheev, 1999 #15)] provides tremendous perspective on the role and benefit of gazetteers within the field of named entity recognition. Gazetteers have often been trumpeted as the key to recognition of entities. There is generally thought to be a relationship between how concise a gazetteer is and the recall and precision of the system. The method proposed here is to use either no gazetteers or small gazetteers with locations of a high frequency of entity names in combination with rule based grammars and maximum entropy statistical models. This is preferred to using large gazetteers. The authors started this work in preparation for the Message Understanding Conference (MUC-7) competition. Each competitor was given 100 articles which were to be annotated. The competition is judged based on recall and precision. Recall is the percentage of correctly annotated tags over the complete list of annotated tags in the solution. Precision measured the percentage of correctly identified tags in the contestant’s submission.

The authors started by building a simple NER tool.  No grammars or language parsing tools were used in this preliminary tool. This tool used the MUC-7 training data provided by the officials of the competition. This collection contained 200 articles. After running the simple NER system on the training data, it gave 1288 person names, 809 names of organisations and 770 names. When run on the MUC-7 test data set (100 articles) it gave a precision of 90% with a corresponding recall between 40 - 70%.

Since the training data was relatively small, a bigger training set was obtained that contained entries from the following sources:

* 5000 locations (country names, states and their biggest cities)
* 33,000 organisation names (obtained from the CIA Fact Book)
* 27,000 names of famous persons.

Using this training set, the tests were again run on the MUC-7 test data set. The authors saw the highest precision for names of locations; that of 94%, followed by person names with a precision of 81% and organization names at 51%. The highest recall was also for location names; this test revealed gave 74%. Person names had a recall of 30% and organization names had only 3%.

The authors then combined both methods and got the following results:

* Recall of 50, precision of 72 for Organizations.
* Recall of 47, precision of 85 for Persons
* Recall of 86, Precision of 90 for Location

The system the authors submitted for the competition uses concepts of internal and external evidence. Internal or phrasal evidence speaks to the evidence that is contained within the named entity itself. There is evidence within the named entity that helps to tell what type of entity it is or whether it is a named entity at all. The phrasal evidence, by its structure aids in the process of classifying the named entity. External or contextual evidence speaks to the information that is contained in the text that helps to tell what type of named entity it is. When there is ambiguity, the general strategy is to decide on a classification only when external evidence or contextual information can be found to confirm the suspected type of classification. Also, in the scenario where a named entity is used once as a particular type of classification, then later used in a different classification, for example, “Adam Kluver” may refer to the company first, then refer to the person later. The approach adopted here is to again rely on contextual information to determine the classification. If no suitable context is found, then the system can then turn to phrasal grammars that can also aid in classification. One such example is where a grammar may dictate that a company ends with “Ltd”.

A four (4) step algorithm was devised for the tagging of named entities.

* Sure-fire rules - for this stage, the text is first POS tagged and tagged with simple semantics. Then a set of phrasal and contextual rules are applied only if the context confirms the phrasal matches. These rules outline basic patterns for the different categories of named entities. One example of a rule given was “Xxxx+ is? JJ\* PROF” where “Xxxx+” represents a sequence of capitalized words, “is?” represents the presence of the word “is”, JJ\* represents zero or more adjectives and PROF represents a profession. One matching sentence would be “Portia Simpson, a former member of parliament”. The rules as they applied, are not stringent in their assignment. For example, “Washington” would only be tagged as a location if it was preceded by a term such as “near”. Named entities are much more likely to remain untagged if there is a disagreeing context at this stage.
* Partial Match 1 - in this stage, the entities tagged thus far are collected and their occurrences in the document are analyzed. This is done first by breaking each named entity generating all the permutations of partial orders of the entity, preserving the order and finding their occurrences within the text. Each occurrence is marked as being the same type of classification as the original. This marking is still not definite as these assumptions may be incorrect since it is possible for the named entity to be used later in the text under a different meaning. This information is then passed on to a pre-trained maximum entropy model which considers other information such as position of the entity in the sentence, the case used originally and elsewhere in the document. If this model confirms the previous assignment, the system marks the entity as a definite match.
* Rule Relaxation - in this stage, the system re-applies grammar rules with less stringent measures. It goes through and tags all person names as such if grammar rules in the partial match or sure fire stage hadn’t tagged the entity as being a different type e.g. ORGANIZATION. The system also tries to resolve conjunction uncertainties with organization names. For a company named “China International Trust and Investment Corp”, the system will check if other parts of the conjunction are present in the text. If not, the system will assume that one entity is being referred to. The system also tries to resolve cases of a sequence of capitalized words. In the case of “Suspended Ceiling Contractors Ltd”, for instance, the system will check to see if the modifier “Suspended” is used elsewhere in the text. If it does, then the system will determine that it is the name of an ORGANISATION. The same approach is used for the possessive in the example “Murdoch’s News Corp”. If “News Corp” appears elsewhere in the text, the system will assume that News Corp is the name of an organisation. Finally, in this stage, the system tags all known organisations and locations from lists that the system has access to.
* Partial Match 2 - at this point, the system goes through and tags smaller components of a named entity. For instance the name “Whyte” is tagged as a PERSON if “James Whyte” was previously tagged as a person. A maximum entropy model is also employed in this stage to further help with the disambiguation. One example cited is that different entities separated by a conjunction are generally of the same type.
* Title Assignment - the authors also had to tag entities found in the title. Although the title was in uppercase, the authors were able to tag entities in the title based on matches (partial or full) found in the body of the text.

The author’s submission got a score of 93% for combined recall and precision. The scores for recall for all entity types (organization, person, location) grew at each stage progressed. The Sure-fire rule stage had it lowest recall for Locations (36%) and highest score of 93% in the Partial Match 2 stage for persons. The precision had very high values across all stages, consistently falling in the range 93-99%. Person names had the highest precision with a score of 99%, while locations had the lowest precision; that of 93%. The recall and precision were also very high for the Title Assignment. The highest recall was 95% for persons and locations, while precision had a high of 97% for person names.

The authors then ran their system in the following configurations:

* Full gazetteer
* Limited gazetteer - in this configuration, the authors initialized the system with named entities from 30 of the 100 official MUC articles and allowed the system to build a gazetteer as it interacted with test data.
* Some locations - in this configuration the gazetteer only contained 200 locations of countries and continents.
* No gazetteer - in this configuration, the system is not able to rely on any list of names of entities but is still able to apply grammar rules to the system.

The results show the Full Gazetteer obtaining scores of precision and recall in the range 90-98%, Limited gazetteers range from 87-92% for recall and 90-97% for precision. In the some locations configuration, the recall was in the range 85-90% while the precision varied from 89-97%. When the system was run with some locations, the recall for locations fell to 46%, while organization and persons had scores of 86% and 90% respectively. The precision for locations was also low; a score of 59% was obtained. Organization and people names still scored high values of 85% and 95% respectively.

Based on the results obtained the system still scores above 80% for named entities (except locations) when no gazetteers are used. However, with the addition of a small, but common list of locations, this increases the recall by more than 40%. A noted reason for the low recall in the scenario where no gazetteers were used was due to missing contextual clues in the body of the text.

# Research Design and Methodology

In order to extract the names of Jamaican geographic locations from natural language, a solution will be designed using the Natural Language toolkit for Python and the Pattern toolkit. Python is an interpreted that has a gentle learning curve and has excellent learning resources ([Madnani [8](#_ENREF_8)]). Python is also very modular and has a very strong set of built-in modules and other third party plugins. There are many modules available to process geographic datasets. The Natural Language Toolkit for Python is a well-documented module that houses many techniques for doing Natural Language analysis. The online tutorials are very well written and provide a very solid base for doing simple and more advanced tasks using NLP. Researchers have also written several reviews that provide gentle introductions to beginning work with NLTK. The NLTK toolkit for Python has been selected in order to develop an algorithm that will extract Jamaican geographic locations from unstructured text. The Pattern toolkit for Python ([Smedt and Daelemans [9](#_ENREF_9)]), was developed at MIT. The Pattern toolkit combines functionality from many different Python modules into one easy to use interface. The Pattern toolkit covers functionality such as downloading pages from the internet, parsing results from search engines, crawling web pages, HTML parser and text analysis. The Python language is also a familiar domain and thus represents a natural choice for the development of the solution.

## Building a corpus

All research reviewed has underlined the importance of training the system properly so that context rules may be derived and the system may be relevant for the domain of text it will process. In particular [David A. Smith [4](#_ENREF_4)] notes that proper training is necessary for the derivation of context rules. Currently, there is no known corpus that contains confirmed references to geographic locations in Jamaica. Therefore, the first step in the research will be to construct a corpus that will contain such references. A local corpus is needed so that oddities and differences for the Jamaican context can be properly understand and integrated into the algorithm for recognizing places. Primarily, a Python script will be built to extract articles from the following websites:

* Jamaica Gleaner
* Jamaica Gleaner Archives

The aim is to obtain enough articles that will contain up to 1000 references to different geographic names across the island. These geographic features are expected to cover parishes and communities. After the information is downloaded using a Python script, each of these articles will be manually geo-tagged. This corpus will be used to train the system. Statistical methods will be used to store the different variations used in referencing these places. Careful attention will be placed on honorifics, capitalization or other cases used in references, name declensions, aliases and geographic distances.

## Building a gazetteer

Approaches have been presented that do not use gazetteers for the process of NER ([David A. Smith [4](#_ENREF_4" \o "David A. Smith, 2001 #7)]), however considering the results of the previous research and the abundance of methods that favoured the use of a gazetteer we have decided to build a gazetteer for the purpose of name recognition. Unfortunately, only international gazetteers currently exist, thus we will use the Global discovery gazetteer and extend it to include geographic locations in Jamaica. A proper source of information for such has not yet been secured. It is expected that a considerable amount of manual effort will need to be placed on manually obtaining lists of the different features in each category then geocoding these references. If we find that the information is not available, the research will be confined to the list of categories that are readily available.

All items stored in our gazetteer will be assigned a class and weightings assigned appropriately. The approach taken here will mirror the approach in [Bruno Pouliquen and Blackler [5](#_ENREF_5)] in assigned classes and weights. We expect to assign classes and weights as dictated below (lowest class first).

|  |  |  |
| --- | --- | --- |
| **Geographic Name Category** | **Class** | **Weighting** |
| Parish | 1 | 30 |
| Community | 2 | 10 |

**Table 1:** Geographic name categories with class and weight

## Develop an algorithm

An algorithm will be built using the Python NLTK module for the extracting of geographic locations. The algorithm will first seek to construct a list of the lowest classes found in the document which will be deemed the domain(s) of the article. Since the domain of articles is generally expected to be in Jamaica, we expect that actual references to Jamaica may not be present in the article. Hence, the algorithm will start with the lowest class and keep incrementing the class until the domain of the document is determined. We also expect that there may be several references to geographic places sharing the same domain. Hence, the algorithm will start by searching for all references to class zero (0) geographic places, incrementing the class each time no result is found.

Once a domain or list of domains is constructed for the document, the algorithm will continue to search the document for references to higher class geographic names in the document. For each result obtained, a weighting will be calculated for the reference. The weighting will be calculated based on the class weighting, weighting based on statistical similarity to usages in training set and mean geometric distance to the domains and its geographic containment by a domain of the document. A centroid will then be calculated for the references in the document. All geometric features that are more than two standard deviations away from the centroid will be discarded. A new centroid will then be calculated from the remaining set of features.

The algorithm will then loop over each of the non-domain features, keeping a context of up to four geographic references before and after the current one. It will check the corpus to compare the feature’s context with contexts for the same feature in the corpus. A weighting will be assigned based on the similarity to the corpus. After each feature contains its final weighting different heuristics will be applied to determine what should be eliminated.

## Evaluation

For the purposes of evaluation, articles will be selected at random from the Jamaica Gleaner and Jamaica Observer websites (once they are not already contained in the training set). The python solution will be used to properly recognize and classify up to 200 references to geographic names with an F-measure of higher than 0.8. Tests will be done first with a threshold set for the weighting to see what values will be obtained for precision and accuracy. Then we will remove the threshold to see what values of precision and accuracy are required to produce an F-measure of 0.8 or above. Finally, the different weightings will be adjusted to see what effect this will have on the accuracy, precision and by extension, the F-measure.

# References

1. Jeonghee Yi†, T.N., Razvan Bunescu, Wayne Niblack†, *Sentiment Analyzer: Extracting Sentiments about a Given Topic using Natural Language Processing Techniques.* Proceedings of the Third IEEE International Conference on Data Mining, 2003.

2. CAROL FRIEDMAN, P., LYUDMILA SHAGINA, MS, YVES LUSSIER, MD, GEORGE HRIPCSAK, MD, MS, *Automated Encoding of Clinical Documents Based on Natural*

*Language Processing.* J Am Med Inform Assoc, 2004. **11**: p. 392-402.

3. Ravin, Y., *Extracting Names from Natural-Language Text.* 1997.

4. David A. Smith, G.C., *Disambiguating Geographic Names in a Historical Digital Library.* P. Constantopoulos and I.T. Sølvberg (Eds.), 2001: p. 127-136.

5. Bruno Pouliquen, M.K., Ralf Steinberger, Camelia Ignat, Tamara Oellinger, Ken and F.F. Blackler, Wajdi Zaghouani, Anna Widiger, Ann-Charlotte Forslund, Clive Best, *Geocoding Multilingual Texts: Recognition, Disambiguation and Visualisation.* In Proceedings of LREC-2006, 2006.

6. Florian, R., et al., *Named entity recognition through classifier combination*, in *Proceedings of the seventh conference on Natural language learning at HLT-NAACL 2003 - Volume 4*2003, Association for Computational Linguistics: Edmonton, Canada. p. 168-171.

7. Mikheev, A., M. Moens, and C. Grover, *Named Entity recognition without gazetteers*, in *Proceedings of the ninth conference on European chapter of the Association for Computational Linguistics*1999, Association for Computational Linguistics: Bergen, Norway. p. 1-8.

8. Madnani, N., *Getting Started on Natural Language Processing with Python.* ACM Magazine, 2007. **13**(4): p. 5-5.

9. Smedt, T.D. and W. Daelemans, *Pattern for Python.* J. Mach. Learn. Res., 2012. **98888**: p. 2063-2067.