Automatic Extraction of Destinations, Origins and Route Parts from Human Generated Route Directions

Xiao Zhang[1], Prasenjit Mitra[1,2], Alexander Klippel[3], and Alan MacEachren[3]

[1] Department of Computer Science and Engineering
[2] College of Information Science and Technology
[3] Department of Geography
The Pennsylvania State University
xiazhang@cse.psu.edu, pmitra@ist.psu.edu,
{klippel,maceachren}@psu.edu

Abstract. Researchers from the cognitive and spatial sciences are studying text descriptions of movement patterns in order to examine how humans communicate and understand spatial information. In particular, route directions offer a rich source of information on how cognitive systems conceptualize movement patterns by segmenting them into meaningful parts. Route directions are composed using a plethora of cognitive spatial organization principles: changing levels of granularity, hierarchical organization, incorporation of cognitively and perceptually salient elements, and so forth. Identifying such information in text documents automatically is crucial for enabling machine-understanding of human spatial language. The benefits are: a) creating opportunities for large-scale studies of human linguistic behavior; b) extracting and georeferencing salient entities (landmarks) that are used by human route direction providers; c) developing methods to translate route directions to sketches and maps; and d) enabling queries on large corpora of crawled/analyzed movement data. In this paper, we introduce our approach and implementations that bring us closer to the goal of automatically processing linguistic route directions. We report on research directed at one part of the larger problem, that is, extracting the three most critical parts of route directions and movement patterns in general: origin, destination, and route parts. We use machine-learning based algorithms to extract these parts of routes, including, for example, destination names and types. We prove the effectiveness of our approach in several experiments using hand-tagged corpora.

Keywords: driving directions, route component classification, destination name identification, geographic information extraction.

1 Introduction

GIScience is intimately linked with a growing interest in revealing how the human mind understands spatial information [1]. In the end, analyses of spatial information have to be meaningful to humans and as such it is only natural to look into how humans make sense of information about the environment in the first place. One of the most influential revelations that brought together researchers from the cognitive science community and spatial information science is the role that metaphors play in understanding (spatial) information. The work on metaphors by Johnson and Lakoff [2] has lead to the

S.I. Fabrikant et al. (Eds.): GIScience 2010, LNCS 6292, pp. 279–294, 2010.

identification of recurring patterns in human (spatial) lives, their direct and embodied experiences that shape their thoughts and that allow humans to understand abstract concepts that are otherwise hard to grasp. These recurring patterns are referred to as *image schema* [3]. Many researchers across different disciplines have built on this work recognizing the importance of identifying and using information that can be organized from an image-schematic perspective.

One of the most fundamental image schemas is that of a *path* with three distinct elements: origin, the path itself, and the destination (source - path - goal in the original terminology). The importance of understanding this seemingly simple schema is corroborated by the fact that a plethora of articles has been published on this topic across disciplinary boundaries [4] [5] [6]. From a GIScience perspective, last year's best paper award for this conference went to Kurata for his paper on the 9+ - intersection model [7]. This model explicitly adds origins and destinations to the 9-intersection model [8] allowing us to characterize the human understanding of movement patterns more adequately. In follow up articles, this topological characterization of a trajectory in relation to a spatially extended entity (a region) has been applied to interpret movement patterns of individual agents [9]. Hence, understanding where someone or something is coming from (Origin), where someone or something is going to (Destination), and how someone or something got from the Origin to its Destination are essential aspects in characterizing and interpreting movement patterns.

The importance of analyzing movement patterns (as one form of spatio-temporal information) is also reflected in current multinational research programs that bring together researchers that develop tools, methods, and theoretical frameworks for properly modeling and analyzing movement patterns [10]. While the majority of research focuses on coordinate-based information, we are interested in identifying how humans characterize movement patterns linguistically and how to interpret these descriptions automatically. Being able to develop computational systems that interpret spatial language is important for the following reasons (among others):

- Text documents are large in number and developing tools to automatically analyze them is essential to cope with what Miller called the data avalanche [11]. We are not only facing a quantitative data avalanche, but also a qualitative one.
- Automatic understanding of large amounts of text on the World-Wide-Web opens opportunities to collect data and perform (spatial) analysis.
- Spoken dialogue can be transformed into written text. Linguistic communication is the primary way for humans to exchange thoughts, ideas, and information.

More specifically, with the spatial information encoded in origins, destinations, and route parts databases can be built that allow for answering questions such as: How many linguistically coded routes ended in the Washington DC area or what linguistic patterns (e.g. relative or cardinal directions) are people in different regions of the US using.

However, the challenge of automatically analyzing linguistically encoded movement patterns, and specifically route directions, are manifold. Any automatic extraction method must deal with:

- the specifics of documents on the syntactic level (e.g., html code)
- underspecified information and/or missing information
- different linguistic styles, varying semantics, synonyms and their disambiguation.

In this paper, we propose algorithms to automatically identify destinations, origins and route parts from text documents. First we use machine-learning-based algorithms to classify the sentences extracted from route direction documents into route components (destinations, origins and route parts). Then, based on the classification results, our algorithm extracts candidates for destination names. After that, the algorithm re-examines the classification results in order to improve the identification of destinations.

The remainder of the paper is structured as follows. In Section 2, we define important concepts used in this paper and formulate our problem. We review related work in Section 3. In Section 4, we describe our proposed algorithms in detail. Section 5 contains the results of our experiments; they show the effectiveness of our algorithms. In Section 6, we conclude the paper and propose future work.

2 Preliminaries and Problem Formulation

Route directions instruct people how to travel from an origin or an area to a destination. We study direction documents that contain the following route components [12] [13] [14]: **destination**, which is the place a person travels to, often an address or the name of the place; **origin**, which is the place or area a person comes from, often a city, a (cardinal) direction or a highway name; and **instructions** (or route parts; these terms will be used interchangeably in the rest of the paper), which are a set of path segments or route segments a person should follow in order to reach the destination from the origin. Figure 1 gives an example of a driving direction Web page [1]. In this document, the destination is "*Directions to IST*"; one of the origins is "*From University Park Airport:*"; the first instruction for this origin is "*Head southwest on Fox Hill Rd. toward High Tech Rd. (2.4 mi)*". In addition to the route components, direction documents also contain information irrelevant to route directions, such as advertisements and phone numbers. They are called "**other**" in this paper.

Route components and other information are expressed in the form of a complete sentence or a stand-alone phrase. They are referred to as "sentence"s throughout this paper. Given the list of all sentences extracted from a document containing driving directions, the **first task** of our study is to classify the sentences into one of the four categories: (1) destination, (2) origin, (3) instruction or (4) other.

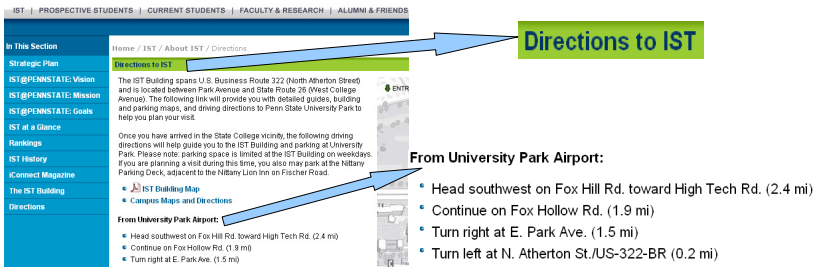


Fig. 1. An example of a direction document

[1] http://ist.psu.edu/ist/iststory/page2.cfm?pageID=1043

A destination is a very important route component because the route ends at the destination and that is where the person following the directions wants to reach. One destination can serve several routes. If the name of the business or organization providing driving directions (referred to as destination names, such as "*Evergreen Golf Club*", "*The Happy Berry, Inc.*") is successfully identified, we can find its coordinates and further locate it on a map. The type of the destination (for example "*university*", "*hospital*") is also very helpful, because if the destination can be narrowed down to a small area and all business names in this area are available, the type can help us pinpoint the destination. Therefore, the **second task** of our study is as follows: given the list of sentences in a direction document, extract the name or the type of the destination.

However, as will be shown in Section 4.2, the recognition of destinations is a difficult problem, especially for stand-alone destination names. Such sentences are often very short. They lack the features that make them stand out from the other route components, and are frequently mis-classified as "other". Without using additional information about the destination names in their context, it will be very difficult to recognize them. Based on our observation, such information can be found in the "**arrival information**" of the directions. The "arrival information" is the last sentence in a set of instructions. The name and type of the destination are often mentioned in it, for example, "*Jordan Hall is located immediately to your right.*". We proposed an algorithm to extract destination names and types from arrival information and use it to improve the recognition of destinations. With this new information, we try to accomplish our **third task:** improve the number of recognized destination sentences over all the destination sentences in the documents (also referred to as *recall* in information retrieval [15]).

3 Related Work

Our system uses machine-learning models to classify sentences extracted from the direction documents into route components. Based on our observation, the sentences displayed a strong sequential nature, for example, instructions are often grouped together; origins are often followed by instructions and other information often appears together. Therefore, we first review related work on machine-learning algorithms designed to label sequential data. Then, we review previous work on sentence classification in other domains. Our task of identifying destination names is related to but essentially different from named entity recognition problems, hence, we review this field too.

3.1 Labeling Sequential Data

Labeling sequential data involves assigning class labels to sequences of observations. Labeling sequential data includes Part of Speech (POS) tagging and entity extraction. Sequential data has: 1) statistical dependencies between the objects to be labeled, and 2) the set of features of the object that can be extracted by observing the object itself. Unlike traditional classification models that make independence assumptions and only model the features within each object, such as Naïve Bayes [16] and Maximum Entropy [17], sequence modeling methods exploit the dependence among the objects. Such methods include Hidden Markov Models (HMMs) [18], Maximum Entropy Markov Models (MEMMs) [19] and Conditional Random Fields (CRFs) [20]. HMMs, based on

a directed graphical model, have been widely used to label sequences. HMMs model the joint probability distribution $p(\mathbf{y}, \mathbf{x})$ where $\mathbf{x}$ represents the features of the objects we observed and $\mathbf{y}$ represents the classes or labels of $\mathbf{x}$ we wish to predict. MEMMs, also based on a directed graphical model, combine the idea of HMMs and Maximum Entropy (MaxEnt). (CRFs) [20] are based on an undirected graphical model. CRFs directly model the conditional distribution $p(\mathbf{y}|\mathbf{x})$. It follows the maximum entropy principle [21] shared by MaxEnt and MEMMs.

3.2 Sentence Classification

Sentence classification has been studied in previous work. Khoo, et al., evaluated various machine learning algorithms in an email-based help-desk corpus [22]. Zhou, et al. studied the multi-document biography summarization problem based on sentence classification [23]. However, in these two approaches, the sentences are treated independently from each other. No interdependencies were considered.

Jindal and Liu studied the problem of identifying comparative sentences in text documents [24]. Such sentences contain two or more objects and the comparisons between them. Their proposed approach uses a combination of class sequential rules(CSR) and machine learning. CSRs are based on sequential pattern mining, which finds all sequential patterns that satisfy a user-specified minimum support constraint. This makes CSRs fundamentally different from our sequential data labeling task.

Hachey and Grover evaluated a wide range of machine learning techniques for the task of predicting the rhetorical status of sentences in a corpus of legal judgements [25]. They examined classifiers making independence assumptions, such as Naïve Bayes and SVM. They also report results of a Maximum Entropy based model for sequence tagging [26]. This approach is similar to MEMMs. However, they evaluate only one sequence labeling model and the features for sentence classification are limited. We have identified a richer set of features that are effective for sentence classification in our domain of interest.

3.3 Destination Name Extraction

Our task of extracting the names of destinations is related to the field of named entity recognition (NER). NER aims to extract entities such as names of persons, organizations, or locations from text. Much work on NER has focused on machine learing methods, such as Maximum Entropy [17] [27], Hidden Markov Model [28], CRFs [20]. However, our task of finding destination names is fundamentally different from the traditional NER task. In driving directions, there are many names. However, most of them are landmarks in the instructions to help travelers locate themselves and make route decisions; only one or a few of them are names of the destination. Additionally, traditional NER methods suffer from the ungrammatical nature of Web pages, such as over-capitalization and under-capitalization [29].

4 Algorithm Description

In this section, we describe our proposed algorithms that classify the route components and identify the destination names in a direction document. Given the list of sentences

(in their original orders) extracted from a document containing route directions, we first use machine-learning algorithms to classify each sentence into one of the four route labels: "DESTINATION", "ORIGIN", "INSTRUCTION" and "OTHER". Based on the classification results, we apply our proposed algorithm to find candidates for destination names from two sources: (1) sentences predicted as a "DESTINATION", or (2) the "arrival information" sentences from the predicted results. Using these extracted candidates, we re-check the sentences labeled as "DESTINATION" and "OTHER" to pick up mis-classified true destinations and finalize the set of candidates.

4.1 Sentence Classification

We use four machine learning models for sentence classification: CRFs, MEMMs, Naïve Bayes, and Maximum Entropy. A list of sentences is extracted from route descriptions using the method described in our previous work [12]. Then, these models use the following features extracted from each sentence to perform the classification:

Bag-of-Words Features: The appearance of each term in a sentence is a feature of the sentence. The same term in different cases are considered to be the same (the case information is captured by the surficial feature discussed next). The algorithms do not eliminate traditional stop words because some of them play an important role in our classification task, for example, "take", "onto" and "at" carry important spatial clues and should not be eliminated; others that do not carry spatial clues do not impact the classifier.

Surficial Features: Surficial features describe the "shape" of the sentences. They are: whether a sentence has only one word, whether a sentence consists of digits only, whether all words have their initials capitalized, and whether all letters are capitalized. We designed this set of features to capture the way certain route components are expressed. For example, the destination and origin may have been emphasized in the document by capitalizing each letter (e.g.,"*DIRECTIONS TO HOSPITALITY HOUSE*", "*FROM THE NORTH*", etc.); phone numbers of the destination, which are all digits, should be labeled as "other" (note that phone numbers can be used to query online phone book services to help identify destinations, however, in this work, we do not consider external knowledge sources and only extract the features within each document).

HTML Visual Features: For direction documents from the Web, different route components have different HTML visual features. HTML authors often use different visual features for different route components. Titles of HTML documents may contain the destination; destinations and origins are often in Headings; links in HTML are often irrelevant to route components. Therefore, we extracted the information about whether a sentence is a title, a link or a heading, as a set of features.

Domain-specific Features: We identified a set of frequent patterns that appear in directions. Such patterns include highway names and particular verb phrases, such as "turn left to ...", "merge onto ..." and "proceed ... miles...". We refer to these patterns as **language patterns**. Using a set of rules consisting of regular expression patterns, we check whether a sentence contains any of these patterns. We designed the set of rules based on an examination of a sample of documents obtained from our collected corpus (Section 5.1 describes how the corpus was built). In our system, we have 24 regular expressions to extract frequent language patterns for instructions, 2 for destinations,

Table 1. Regular Expressions to extract domain-specific features

Feature	Regular Expressions	Example
DEST1	\s*(driving)?\s*(direction|directions)\s+to\s+\w{2,}.*	driving directions to IST
DEST2	.*visiting\s*\w{2,}.*	Visiting Winterthur
ORIG	.*(direction|directions|coming)?\s*(from—via)\s* HighwayPS? (the)?\s? CardinalDirPS .*	coming from I-80 East
INST1	.*turn\s+(left|right).*	turn left
INST2	.*turn \s+off\s+at\s+exit(\s+\d+)?.*	turn off at Exit 168
INST3	.*(proceed|travel|drive|go|turn|head)\s+CardinalDirPS.*	traval north for 3 miles.
INST4	.*(proceed|continue)\s+(along|into|past|on).*	proceed along College Avenue
INST5	.*continue\s+(CardinalDirPS\s+)?.*	continue east
INST6	.*take\s+(?:the\s+)?(?:OrdinalNum\s+)?(\w*\s+)?exit.*	take the second exit
INST7	.*follow\s+the\s+exit.*	follow the exit to State College
INST8	.*follow\s\d{1,5}(\.\d{1,5})?\smile(s)?.*	follow 3.4 miles
INST9	.*follow\s+(the\s+)?sign.*	follow the sign to San Jose
INST10	.*follow\s+(the\s+)?(\w+\s+)*(avenue|street|road).*	follow College Avenue
INST11	.*follow\s+ HighwayPS .*	follow I-80 west
INST12	.*take\s+ HighwayPS [./]*	take US-322/220
INST13	.*exit\s+(at\s+)? HighwayPS .*	exit at PA Ruote 23
INST14	.*exit\s+(left|right)?\s*(at|onto|to)\s+.*	exit right onto KING Rd
INST15	.*stay\s+(on|straight).*	stay on I-80
INST16	.*(make|take)\s(left|right)\s(onto|to|on).*	make left onto Columbus Blvd.
INST17	.*(make|take)\s+(the|a|(the\s+(next|CardinalDirPS))) \s+(\w*\s+)?(left|right).*	take a sharp left
INST18	.*(bear|keep)\s+(to\s+the\s+)?(left|right).*	bear left
INST19	.*merge\s+(left|right)?\s*onto.*	merge onto I-670 E
INST20	.*take.*(exit|ramp|bridge|turnpike|thruway|expressway).*	take NJ Turnpike South to
INST21	.*HighwayPS \s+CardinalDirPS\s+exit.*	to Route 322/22 West Exit
INST22	.*exit\s*\d+.*	take exit 168.
INST23	.*StreetTypesPS\s*CardinalDirPS.*	East 49th Street north
INST24	.*CardinalDirPS .* StreetTypesPS.*	...north Atherton Street...
OTHER1	.*\b[a-zA-Z0-9._%-]+@[A-Z0-9.-]+\.[A-Z]2,4\b	match email addresses
OTHER2	\W+	match lines with only non-word characters

1 for origin and 2 for other. Table 1 gives all the regular expressions and examples of language patterns in the text ("HighwayPS" is the pattern string for highway names; "CardinalDirPS" is the pattern string to match the orientation words such as east and west). Another set of domain-specific features are nouns and noun phrases frequently used to specify the types of destinations, such as "hotel", "restaurant", "campus", etc. We created a **dictionary** of such nouns and noun phrases referring to a place or a location. Whether a sentence contains a dictionary entry is a binary feature of this sentence.

Window Features: are specified language patterns that appear in a specific window around a sentence. Window features are extracted after the surficial and language pattern features have been extracted. Our system checks whether specified features or any one of a set of specified features exists in the previous and/or the following sentences of the current sentence. One window feature checks if the surrounding sentences match any of the 24 regular expressions for instructions. Since instructions are often grouped

together, the current sentence may also be an instruction if it has this window feature. Consider the following three consecutive sentences extracted from the document shown in Figure 1: (1) *"Head southwest on Fox Hill Rd. toward High Tech Rd. (2.4 mi)"*, (2) *"Continue on Fox Hollow Rd. (1.9 mi)"* and (3) *"Turn right at E. Park Ave. (1.5 mi)"*. When we examine the second sentence, we can see that sentence (1) and (2), match language pattern INST3 and INST1 (see Table 1), respectively. Therefore, sentence (2) has the window feature we described above. Other window features are: whether there is a street name and number in the previous/following two sentences, whether there is a zip code in the following two sentence.

4.2 Destination Name Recognition and Re-classification

Deficiencies in Sentence Classification Results. The machine-learning models give us the list of sentences with predicted route component labels. After analyzing the classification results, we found that the classification accuracy for destination sentences was not satisfactory (see Section 5.2). Although the best precision for destinations can reach about 80%, the recall is about 40% for all MEMMs and CRFs. Since destinations are a very important route component in descriptions of directions, the recognition of destinations from the list of sentences is crucial.

Based on our analysis, we noticed that many destination sentences were classified as "OTHER" by mistake (the reasons will be given in Section 5). Such destination sentences often contain the name of the destination. Their features are very similar to the features displayed by "OTHER" sentences. For example, the name of the business or institute may stand alone as an individual sentence and not have the strong features associated with destinations. Such sentences are often very short and do not have many language pattern features to extract. The terms in the name are often capitalized as they are in "OTHER" sentences. Sometimes abbreviations are used for the destination name, which makes it more similar to entries classified as "OTHER". Without looking for more information from the context, it will be very difficult to identify such destination sentences with destination names with high accuracy.

Luckily, when people create route descriptions, they sometimes put destination names or the type of the destination in the arrival information of a set of route instructions. The names are often embedded in an obvious language pattern. We show some examples of the mis-classified destination sentences and the contextual information we can utilize to improve classification below:

Example 1:
Destination: Trial Lawyer Harford Connecticut - Levy and Droney P. C.
Arrival Info.: Levy and Droney is located in the building on the right.
Example 2:
Destination: 2002 - 2005 Atlantic Country Club
Arrival Info.: Atlantic Country Club will be 2 miles down on the left.

Once the destination name has been identified from the arrival information, we can use the names to match the names in the sentences predicted as "OTHER" and re-classify them.

In addition, the destination names in the sentences classified as "DESTINATION" can also be used to find the mis-classified destination sentences. Given the high

precision of predicted destination sentences, it is reasonable to assume that most destination names can be found in the sentence already predicted as "DESTINATION". Once the names are extracted, we can again use them to identify mis-classified destinations. Here are some examples of such cases. In each one of them, the mis-classified sentence is a destination but classified as "other".

Example 3:
Correctly Identified: Delaware Court Healthcare Center - Map (Title)
Mis-classified: Delaware Court Healthcare Center (at the end of the document)
Example 4:
Correctly Identified: Directions to NIBIB (at the beginning)
Mis-classified: About NIBIB (at the end of the document)

In Example 3, since being in the title is a strong indicator of being a destination, the first sentence is classified correctly. However, the second one was not recognized since it is surrounded by "other". The dependency assumption caused the classifiers to make the wrong decision. The reasons are the same for Example 4.

Destination Name Extraction. We take the output of the sentence classifiers (a list of sentences with predicted route component labels for each one) and find blocks of instructions. Then the last sentence, or arrival information, of each block is extracted. Since destination names are proper nouns or proper noun phrases and destination types are a relatively small set of nouns, we rely on a Part-of-Speech tagger [2] and the dictionary of type nouns (introduced in Section 4.1) to identify the nouns and proper nouns in these sentences. However, not all proper nouns are destination names; we have to filter the results. Arrival information often has language patterns and the destination names are at certain positions of such pattern. Consider the sentence: "*The church is on your left.*". When the pattern "*... is on your left*" is identified, the nouns in the front of it are very likely to be the destination name or type, thus the range for searching for the destination names or types is narrowed from the entire sentence to a portion of it. We created a set of regular expressions to represent such patterns and specified the places where the destination names and types can be found in each regular expression. Our algorithm matches the arrival information sentences against these patterns. If it finds a match, the algorithm only focuses on the nouns and proper nouns in part of the sentence. If not matched, we use a heuristic rule to filter the results: the noun or noun phrase must have at least one proper noun or one type noun in it, otherwise it is discarded. After that, we further remove the nouns and noun phrases that are able to pass the above filtering steps and are preserved, but are not destination names such as "*parking lot*"s or "*parking garage*"s without names. We remove them from the returned results. Algorithm 1 gives the details. The same algorithm can be also applied to the sentences predicted as "destination" to extract the names or types from them, except that a different set of regular expressions are used. In our actual system, both "DESTINATION"s and "ARRIVAL INFORMATION" are used as the input to this algorithm and it returns a set of candidate destination names and types.

Improving Recognition of Destinations. After extracting possible destination names and types from arrival information and predicted destination sentences, we use them

[2] In our system, we used LingPipe POS tagger: http://alias-i.com/lingpipe/

Algorithm 1. Extract Possible Destinations from Arrival Information

Input:
(1)a sentence s (2)a trained POS tagger T (3)a dictionary of type nouns D (4)a set of patterns P
Output:
a set of possible destination names $Dest$
Procedure:

```
1: Dest ← ∅;
2: find all nouns and noun phrases in s using T;
3: if s matches any pattern in P then
4:     find all nouns and noun phrases in the capturing group and insert into n;
5: else
6:     for each noun or noun phrase n in s do
7:         if n has at least one type noun in D or a proper noun then
8:             insert n into Dest;
9:         end if
10:    end for
11: end if
12: remove fake destinations from Dest;
13: return Dest;
```

to re-examine the sentence predicted as "OTHER". During the re-examination, we also filter the extracted destination names and types to generate the final set of possible destination names and types to return to the users. We first pick all the type nouns in the set generated from Algorithm 1 and search for them in the sentences labeled as "DESTINATION" and "OTHER". If this type noun appears in the sentence, we find the proper nouns in front of this type noun and consider the whole noun phrase as one of destination names. Then we change the label to "DESTINATION". After processing the type nouns, we match proper names against the sentences. Again, if this proper name is found in the sentence, we consider it as one of the destination names and change the label to "DESTINATION". At the end, the re-classified sentences and the set of destination types and names will be returned to the user (see Algorithm 2).

5 Experiment Results

In this section, first, we describe how we built our data set. Then, we give the sentence classification results for different machine-learning models. The results for destination-name extraction and re-classification are reported afterwards.

5.1 Data Set

We identified a set of over 11,000 web pages containing route directions using the search results of the Yahoo![3] search engine. The search engine was queried with a set of carefully selected keywords such as "direction, turn, mile, go", "turn, mile, follow, take, exit" etc. since these phrases are typically present within documents containing

[3] www.search.yahoo.com

Algorithm 2. Re-classification and Destination Name Extraction

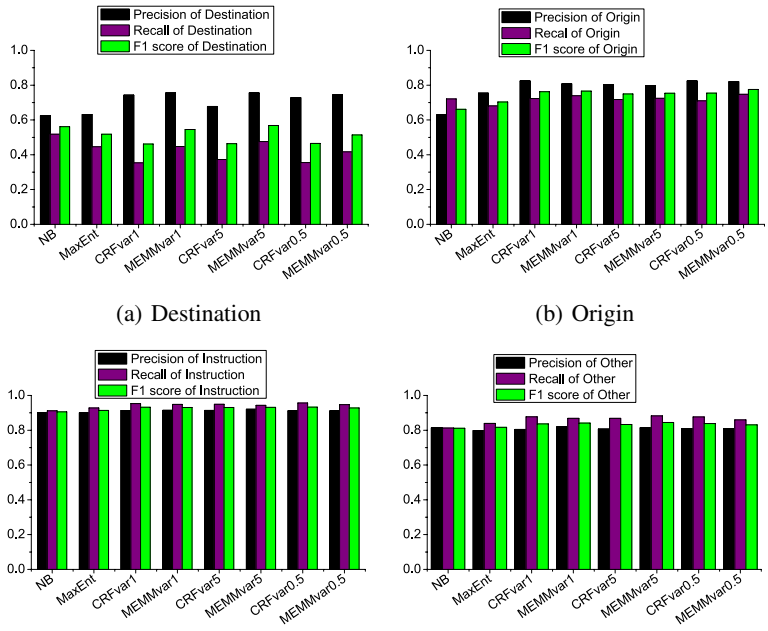
Input:
(1) L: a list of sentences with predicted route component labels
(2) S_{cand}: a set of proper names or types extracted from predicted destinations and arrival information.
Output:
(1) L': a new list of sentences with route components labels
(2) S_{dest}: a set of destination names or type
Procedure:

```
1: S_dest ← ∅, L' ← L;
2: for each sentence l labeled as "destination" or "other" in L' do
3:    for each type noun t in S_cand do
4:       if l contains type noun t then
5:          extract the proper name n in front of t in l, insert n into S_dest;
6:          change the label of l to be "destination";
7:       end if
8:    end for
9:    for each proper name p in S_cand do
10:      if l contains p as a substring then
11:         insert p into S_dest;
12:         change the label of l to be "destination";
13:      end if
14:   end for
15: end for
16: for each sentence l labeled as "destination" in L' do
17:    extract proper names and type nouns in l and insert into S_dest;
18: end for
19: return L' and S_dest;
```

route directions. Each set of queries contains 4 to 7 keywords. Manual examination shows 96% of these documents contain route directions.

5.2 Sentence Classification Results

We evaluated four machine-learning models. Two of them are sequence-labeling models (CRFs and MEMMs). In order to examine the sequential dependency of route components, we use two other models based on the independence assumption: Naïve Bayes and Maximum Entropy models. For CRFs and MEMMs, we tried different values for the initial Gaussian Variance: 1.0, 5.0 and 0.5. The set of sentences we used to train and test the models contains over 10,000 hand-labeled sentences extracted from 100 HTML documents. We used a 10-fold cross validation technique to evaluate these models. Note that in order to preserve the sequential natural of the sentences, we divided this data set by documents, not by individual sentences. As shown in Figure 2, MEMMs and CRFs outperform Naïve Bayes and Maximum Entropy. The recognition of "Origin", "Instructions" and "Other" are reasonable. However, recognizing "Destination" is a hard problem. "Destination" sentences are often very short and lack effective features. This is the reason we utilized the arrival information to find the destination names.



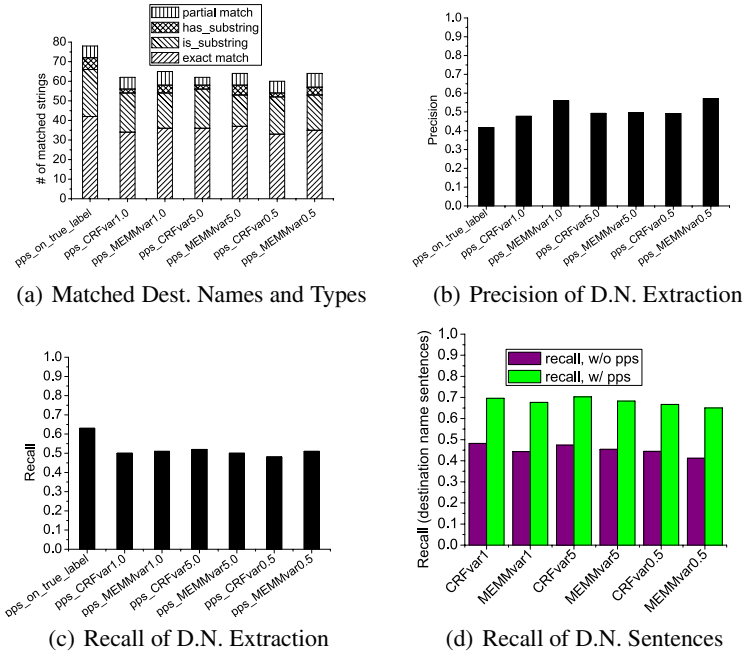
(a) Destination (b) Origin

(c) Instruction (d) Other

Fig. 2. Detailed Analysis of Each Class

5.3 Destination Name Extraction

We applied our proposed algorithms as a post-processing step (PPS) after we obtained the sentence classification results. First, we evaluate the performance of destination name and type extraction. We picked 46 of the 100 documents and hand-labeled a total of 100 destination names and types. For example, in one document, the destination is called “Berkeley campus”. It is also referred to as “campus” or “university” in other parts of the document. All three are hand-tagged as destinations. This set of destination names is our ground truth. PPS is applied to the CRFs and MEMMs classified sentences, as well as hand-labeled sentences, to extract destination names and types. We do so to evaluate how well the PPS performs without the noise introduced by the classifiers. We consider the following matching schema between the extracted names and types: the extracted name (1) has an exact match in the ground truth; (2) is a substring of an entry in the ground truth; (3) contains an entry in the ground truth as a substring; (4) is a partial match (they share common terms but do not have substring relationship); or (5) does not match at all. Figure 3(a) shows the number of names/types in different matching schema. As we have expected, PPS on true labels yields the highest number of exact matches. PPS on MEMMs with variance 5.0 and 1.0 gives the best exact matches and total matches among different classifiers. The precision is the number of matched names divided by the total number of names retrieved by the PPS. Recall is the number of correctly retrieved **unique** names and types in the ground truth divided by the total number of names in the ground truth. Note that when calculating recall, the same name or type that appeared multiple times is counted only once. Due to this reason, precision



(a) Matched Dest. Names and Types (b) Precision of D.N. Extraction

(c) Recall of D.N. Extraction (d) Recall of D.N. Sentences

Fig. 3. Extraction of Destination Names and Types

and recall are *NOT* calculated on the same objects. Therefore, no F1 score (a weighted average of the precision and recall) is given here. Figure 3(b) and 3(c) gives the details.

5.4 Recognition of Sentences with Destination Names

The extracted destination names and types from the classification results are then used to re-classify the sentences labeled as "Other". Figure 3(d) shows that the recall of sentences containing destination names or types has been improved substantially in comparison to the classification results without PPS; this proves the effectiveness of the PPS module. Note that in the destination sentences, there are also addresses or parts of addresses. We have not counted them in the calculation of the recall. Calculating the precision of destination name sentences requires counting the false positives (sentences labeled as "destination" by mistake). However, this number is hard to get because it is not easy to tell which mistake is introduced by the addresses.

5.5 Further Discussions

We also analyzed the destination sentences that are not predicted correctly and not recognized by the post-processing algorithm. We found that some destination names are stand-alone sentences without strong features and thus mis-classified. What made the problem even harder is that, in the arrival information the author of the route description used another name for the same destination. Therefore, they are not picked up in the post-processing step. Here is an example: at the beginning of the document, the

destination is mentioned as “Delaware Academy of Medicine”. The phrase itself is a “sentence” (for our purposes). In the arrival information, the destination is mentioned as “Christiana Hospital campus”. Our post-processing algorithm successfully recognized this phrase and considered it as a candidate for destination names. However, linking the two names together requires external knowledge. We do not have the ability to do such linking now, but will integrate external knowledge, such as that obtained from a geographic database, into the system in the future.

6 Conclusions and Future Work

All movement patterns of individual entities consist of three distinct parts: origins, destinations, and the path/route. The importance of understanding and identifying these parts has been recognized in different disciplines and has a central place in interpreting movement patterns from a cognitively relevant spatial perspective. While quantitative approaches are needed to handle the avalanche of data that becomes available through the development of spatial (tracking) technologies, we focus on the equally important problem of developing tools and methods to handle the qualitative data avalanche. Web documents or transcriptions of verbal communications are some of the most important sources of spatial information, and spatial language is one of the most challenging to handle automatically. We discussed our work toward automatic extraction of route components from documents containing route directions. We focus primarily on the extraction of destination names and types and on improving the recognition of destination sentences. Our system uses trained machine-learning models to classify sentences into route components. Based on the results of this classification, we proposed new algorithms that extract possible destination names and types from certain parts of the document. With this new knowledge, the system checks the re-occurrence of the extracted names and types to find misclassified destinations so that the recall is improved. Experiments have shown the effectiveness of our new methods.

A future step is to query various external information sources, such as online phone book services, web search engines and digital maps, and combine the returned results, in order to identify and geo-code the destinations. Another challenge is the occurrence of multiple destinations and origins that may appear in one document and that are sometimes not well ordered. Thus, finding the correct association of destinations, origins and instructions to form a complete route will be our next step. Additionally, we will also work on matching direction descriptions to GIS databases and geographic ontologies to support both disambiguation and enable human interpretation and refinement.

Acknowledgement

Research for this paper is based upon work supported National Geospatial-Intelligence Agency/NGA through the NGA University Research Initiative Program/NURI program. The views, opinions, and conclusions contained in this document are those of the authors and should not be interpreted as necessarily representing the official policies or endorsements, either expressed or implied, of the National Geospatial-Intelligence Agency, or the U.S. Government.

References

[1] Mark, D.M., Frank, A.U. (eds.): Cognitive and linguistic aspects of geographic space. Kluwer, Dodrecht (1991)

[2] Lakoff, G., Johnson, M.: Metaphors we live by. University of Chicago Press, Chicago (1980)

[3] Johnson, M.: The body in the mind: The bodily basis of meaning, imagination, and reasoning. University of Chicago Press, Chicago (1987)

[4] Allen, G.: Principles and practices for communicating route knowledge. Applied Cognitive Psychology 14(4), 333–359 (2000)

[5] Pick, H.: Human navigation. In: Wilson, R.A., Keil, F.C. (eds.) The MIT encyclopedia of the cognitive sciences, pp. 380–382. MIT Press, Cambridge (1999)

[6] Talmy, L.: Fictive motion in language and "ception". In: Bloom, P., Peterson, M.P., Nadel, L., Garrett, M.F. (eds.) Language and space, pp. 211–276. MIT Press, Cambridge (1996)

[7] Kurata, Y.: The 9+-intersection: A universal framework for modeling topological relations. In: Cova, T.J., Miller, H.J., Beard, K., Frank, A.U., Goodchild, M.F. (eds.) GIScience 2008. LNCS, vol. 5266, pp. 181–198. Springer, Heidelberg (2008)

[8] Egenhofer, M.J., Herring, J.R.: Categorizing binary topological relations between regions, lines, and points in geographic databases: Technical Report, Department of Surveying Engineering, Univeristy of Main (1990)

[9] Kurata, Y., Egenhofer, M.J.: Interpretation of behaviors from a viewpoint of topology. In: Gottfried, B., Aghajan, H. (eds.) Behaviour monitoring and interpretation. Ambient intelligence and smart environments, pp. 75–97 (2009)

[10] `http://www.cost.esf.org/domains_actions/ict/Actions/IC0903-Knowledge-Discovery-from-Moving-Objects-MOVE-End-date-October-2013`

[11] Miller, H.J.: The data avalanche is here. Shouldn't we be digging? Journal of Regional Science (in press)

[12] Zhang, X., Mitra, P., Xu, S., Jaiswal, A.R., Klippel, A., MacEachren, A.: Extracting Route Directions from Web Pages. In: WebDB 2009 (2009)

[13] Golledge, R.G.: Human wayfinding and cognitive maps. In: Golledge, R.G. (ed.) Wayfinding behavior. Cognitive mapping and other spatial processes, pp. 5–45 (1999)

[14] Denis, M., Pazzaglia, F., Cornoldi, C., Bertolo, L.: Spatial discourse and navigation: An analysis of route directions in the city of Venice. Applied Cognitive Psychology (1999)

[15] Manning, C.D., Raghavan, P., Schüze, H.: Introduction to Information Retrieval. Cambridge University Press, Cambridge (2008)

[16] Lewis, D.D.: Naive (bayes) at forty: The independence assumption in information retrieval. In: Nédellec, C., Rouveirol, C. (eds.) ECML 1998. LNCS, vol. 1398, pp. 4–15. Springer, Heidelberg (1998)

[17] Borthwick, A.: A maximum entropy approach to named entity recognition. Ph.D. thesis, New York University (1999)

[18] Freitag, D., McCallum, A.: Information extraction using hmms and shrinkage. In: AAAI Workshop on Machine Learning for Information Extraction (1999)

[19] McCallum, A., Freitag, D., Pereira, F.: Maximum entropy markov modes for information extraction and segmentation. In: Proceedings of ICML (2000)

[20] Lafferty, J., McCallum, A., Pereira, F.: Conditional random fields: Probabilistic models for segmenting and labeling sequence data. In: Proceedings of ICML (2001)

[21] Berger, A.L., Pietra, S.A.D., Pietra, V.J.D.: A maximum entropy approach to natural language processing. In: Computational Linguistics (1996)

[22] Khoo, A., Marom, Y., Albrecht, D.: Experiments with sentence classification. In: ALTW (2006)

[23] Zhou, L., Ticrea, M., Hovy, E.: Multi-document biography summarization. In: Proceedings of EMNLP (2004)
[24] Jindal, N., Liu, B.: Identifying comparative sentences in text documents. In: Proceedings of SIGIR, pp. 244–251 (2006)
[25] Hachey, B., Grover, C.: Sequence modelling for sentence classification in a legal summarisation system. In: Proceedings of 2005 ACM Symposium on Applied Computing (2005)
[26] Ratnaparkhi, A.: A maximum entropy part-of-speech tagger. In: EMNLP (1996)
[27] Klein, D., Smarr, J., Nguyen, H., Manning, C.D.: Named Entity Recognition with Character-level models. In: CoNLL-2003, pp. 180–183 (2003)
[28] Bikel, D.M., Schwartz, R.L., Weischedel, R.M.: An algorithm that learns what's in a name. Machine Learing 34(1-3), 211–231 (1999)
[29] Ding, X., Liu, B., Zhang, L.: Entity Discovery and Assignment for Opinion Minig Applications. In: KDD 2009 (2009)