Extracting Route Directions from Web Pages

Xiao Zhang[*], Prasenjit Mitra[*†], Sen Xu[‡], Anuj R. Jaiswal[†],
Alex Klippel[‡], Alan MacEachren[‡]
[*]Department of Computer Science and Engineering
[†]College of Information Sciences and Technology
[‡]Department of Geography
the Pennsylvania State University

xiazhang@cse.psu.edu, {pmitra, ajaiswal}@ist.psu.edu, {senxu, klippel, maceachren}@psu.edu

ABSTRACT

Linguists and geographers are more and more interested in route direction documents because they contain interesting motion descriptions and language patterns. A large number of such documents can be easily found on the Internet. A challenging task is to automatically extract meaningful route parts, i.e. destinations, origins and instructions, from route direction documents. However, no work exists on this issue. In this paper, we introduce our effort toward this goal. Based on our observation that sentences are the basic units for route parts, we extract sentences from HTML documents using both the natural language knowledge and HTML tag information. Additionally, we study the sentence classification problem in route direction documents and its sequential nature. Several machine learning methods are compared and analyzed. The impacts of different sets of features are studied. Based on the obtained insights, we propose to use sequence labelling models such as CRFs and MEMMs and they yield a high accuracy in route part extraction. The approach is evaluated on over 10,000 hand-tagged sentences in 100 documents. The experimental results show the effectiveness of our method. The above techniques have been implemented and published as the first module of the GeoCAM[1] system, which will also be briefly introduced in this paper.

1. INTRODUCTION

Descriptions of motion, such as route directions in text corpora provide important information and have fascinated researchers for a long time. Since 1970s, linguists and geographers have used route directions to study human spatial cognition, geo-referencing, analyzing route characteristics and building databases of linguistically characterized

[1]Geographic Contextualization of Accounts of Movement. http://cxs03.ist.psu.edu:8080/GeoCAMWeb/

movement patterns [8]. As web technology thrives, a large number of route direction documents have been generated and are available on the Internet. A business, organization or institution usually provides human-generated direction information on its web site to give instructions to travellers from different places to arrive there. Such direction web pages contain both meaningful route parts as well as additional contents irrelevant to finding ones way (e.g., advertisement, general descriptions). Although humans mostly manage to follow these route directions, such manual techniques do not scale to a large corpora of documents. Dealing with real-world corpora requires a scalable information system that can automatically detect and extract route directions in web pages. A challenging task in building such a system is to extract meaningful route parts, namely destinations, origins and instructions (or actions) from contents other than route directions.

In addition to linguistic use, such human-generated route direction information, if extracted, can be used as supplementary information for auto-generated directions, such as Google Maps [2]. Human-generated directions frequently use more obvious landmarks than street names as decision points, for example, "turn right at the McDonald's", which are more helpful for users. They also provides additional information for correction or re-direction, for example, "if you see the school, you've gone too far". Such information is important for users to find their ways but is currently missing in auto-generated directions. Once the route directions in human-generated texts can be extracted, they can be incorporated with map products and better serve the users.

In route direction web pages, **destination** refers to the location where the route ends, usually the business, organization or institute hosting the web site, e.g. "Directions to the Campus". **Origin** specifies the starting point of the route and helps travellers to choose which set of instructions they should follow in order to arrive to the destination, e.g. "From New York". **Instructions** are a set of actions to follow at specified landmarks or decision points such as highways or intersections, e.g. "Merge onto US-220 S toward US-322". In direction web pages, route parts are expressed in the form of a complete sentence, an independent phrase or a single word. We will refer to them as "sentence" in the rest of the paper. The automatic route part extraction proposed in this paper has the goal to classify sentences into

[2]maps.google.com

one of four classes: 1) destination, 2) origin, 3) instruction or 4) other.

The first task of route parts extraction is to delimit sentences in HTML documents. The difficulty of this task is that HTML authors frequently uses HTML structural, positional and visual features, such as columns and table items, as indicators of sentence boundaries and omit traditional sentence boundary indicators, such as punctuation marks, capitalization of the initial word in sentence and abbreviations. Sunayama et al.[16] proposed a method to utilize HTML tags and period mark to extract sentences from Japanese web pages. However, the approach they proposed is not suitable for English language or route parts (details will be discussed in Section 3). We propose an alternative algorithm which utilizes both the HTML tag information and natural language knowledge to delimit sentences from HTML documents.

After the sentences are extracted from HTML documents, the second task is to classify them into one of the four classes mentioned above. Previous work [7] [6] examined classification models based on independence assumptions such as Naive Bayes [10] and Maximum Entropy [2]. They assume the sentences are independent from each other. However, in our scenario, the route part sentences display a strong sequential nature. For example, a destination is usually followed by an origin; an origin is usually followed by a set of instructions and instructions are usually grouped together. Based on this observation, we propose to use sequence labelling models such as Conditional Random Fields [9] and MEMMs [11] for sentence classification. These models consider the inter-dependencies between route part sentences and improve classification accuracy.

The contributions of of this paper can be summarized as follows:

- We build a system to automatically identify route parts in web pages containing route directions. To the best of our knowledge, there is no published work on this problem.

- We study the dependencies between direction sentences in a document and propose to use sequence labelling models for sentence classification. Different classification models are evaluated and compared.

- We study the impact of different feature sets on the sentence classification performance, including language patterns, HTML visual features, dictionaries, etc.

- We propose an alternative approach of an existing work for sentence extraction directly from HTML documents. Our approach fits better to English language and route description documents.

The rest of the paper is organized as follows: Section 2 gives problem definitions. Section 3 reviews previous work. Section 4 presents details of our proposed sentence delimitation algorithm, models for sentence classification and various features. Experiment results are given in Section 5. Section 6 briefly introduces the route parts extraction module of our GeoCAM system and Section 7 concludes the paper and shows future work.

2. PROBLEM ANALYSIS AND DEFINITION

Businesses and organizations usually provide driving directions on their web sites. Such direction pages contain the following contents:

Destination: The place to which a person travels, usually an address or the name of the place.

Origin: The place or region a person comes from, usually a city, an orientation (such as North and South), or a highway name.

Instruction: A set of actions a person should follow in order to reach a specified destination from an origin.

Other: Any contents other than the above three route direction parts, such as phone numbers or advertisements.

Route parts are carried by complete sentences, phrases or single words. Given an HTML document containing route directions, the first step of route parts extraction is to find the objects to classify - the sentences. Sentence delimitation in HTML is different from the delimitation in plain text. First, HTML authors frequently use HTML structural, positional and visual features to indicate sentence boundaries, instead of punctuation marks. For example, a sentence may be bounded by columns in tables. Second, when converting an HTML document into a plain text document, text pieces belonging to different sentences can be concatenated. Thus converting HTML document to plain text and then using existing sentence delimitation tools (e.g. LingPipe[3]) will fail to successfully extract sentences. Moreover, HTML tags, such as $<B>$ and $<A>$, break a sentence into pieces. Therefore, using tags to delimit sentences will not be accurate.

The above problems happen because sentence boundary information in directions generated by humans uses both HTML tags and natural language rules inconsistently. Thus an effective sentence delimitation algorithm should take into consideration both the HTML tags and natural language knowledge. We propose such an algorithm in Section 4.1 and define the problem below:

Definition 1 (HTML Sentence Delimitation). *Given an HTML document, HTML sentence delimitation is to delimit the sentences carrying complete and independent route parts information in HTML source code.*

Sentences extracted from HTML will be further assigned route parts class labels by the sentence classifier. We define the route parts classification problem as follows:

Definition 2 (Route Parts Classification). *Given the list of sentences extracted from an HTML document containing route directions, the Route Parts Classification task is to accurately assign each sentence the following class labels: destination, origin, instruction or other.*

3. RELATED WORK

3.1 Labelling Sequential Data

Labelling sequential data is a task of assigning class labels to sequences of observations. Application of labelling sequential data includes Part of Speech (POS) tagging and entity extraction. Sequential data has two characteristics:

---

[3]http://alias-i.com/lingpipe

1) statistical dependencies between the objects we want to label, and 2) the set of features contained by the object itself. Unlike traditional classification models that make independence assumptions and only model the features within each object, such as Naïve Bayes [10] and Maximum Entropy, sequence modelling methods exploit the dependence structure among the objects.

Graphical models are a natural choice for labelling sequences. Hidden Markov Models (HMMs) [3] [13], based on a directed graphical model, have been widely used in labelling 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. Another approach based on a directed graphical model, Maximum Entropy Markov Models (MEMMs) [11], combines the idea of HMMs and Maximum Entropy (MaxEnt) [2]. Conditional Random Fields (CRFs) [9] are based on an undirected graphical model, thus avoids the label-bias problem [9]. CRFs directly model the conditional distribution $p(\mathbf{y}|\mathbf{x})$. It follows the maximum entropy principle [1] shared by MaxEnt and MEMMs. CRFs have been successfully applied to many applications such as text processing [12] and chemical entity recognition [15].

3.2 Sentence Classification

Sentence classification has been studied in previous work in different domains. Khoo et al., evaluated various machine learning algorithms in an email-based help-desk corpus [7]. Zhou et al., studied the multi-document biography summarization problem based on sentence classification [17]. However, in the two works, 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 [6]. Their proposed approach is a combination of class sequential rule (CSR) and machine learning. CSR is based on sequential pattern mining, which is to find all sequential patterns that satisfy a user-specified minimum support constraint. That makes CSR fundamentally different from our sequential data labelling 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 [4]. 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 [14]. This approach is similar to the framework of MEMM. However, only one sequence labelling model is evaluated and the features for sentence classification are limited. We identified richer sets of features that are effective for sentence classification.

3.3 Sentence Extraction from HTML

Sunayama et al.[16] proposed an approach to extract sentences from HTML documents in order to solve the web page summarization problem. They used block-level tags, link tags ($<$A$>$) and period mark to segment text. Then they rearrange the text pieces by putting small pieces together to guarantee the length of a sentence. However, we discovered that the rearrangement in this paper will be disastrous for route parts because destinations and origins are usually short independent phrases, thus should not be concatenated

Tags to concatenate	Tags to skip
STRONG	SCRIPT
I	STYLE
FONT	OBJECT
EM	OPTION
B	IMG

Table 1: HTML tag examples

to other text. Besides, we assume that the text should be separated by HTML tags except inline-tags and treat $<$A$>$ tags differently. We found that $<$A$>$ tags should be used to concatenate, instead of segmenting, adjacent text pieces. We also considered more natural language knowledge, such as abbreviations and punctuation marks other than period.

4. ROUTE PARTS EXTRACTION

4.1 Sentence Extraction from HTML

Based on the observation that sentence boundaries are indicated by natural language knowledge together with visual and structural features introduced by HTML tags, we propose an algorithm which utilizes indicators from both sides in sentence delimitation.

Our approach first converts an HTML document into a DOM tree and then traverses the tree in a depth-first order. The text nodes encountered will be stored in a list of text except for the following two cases: if text is a child node of a tag node in a pre-defined tags-to-skip list (this list contains tag nodes of which the text children will not be visible when the HTML is rendered by the browser), the algorithm skips this text node; if two text nodes are separated by a tag in the pre-defined tags-to-concatenate list (this list contains the tags which do not indicate sentence boundaries), then the two text pieces are concatenated and put in the list of text. Then the algorithm uses natural language knowledge to further segment each text piece into sentences. Table 1 gives some examples in the tags-to-concatenate list and the tags-to-skip list, and Algorithm 1 shows the details.

4.2 Feature Set

Various sets of features have been extracted for machine learning models for the sentence classification task. Our feature sets can be categorized as follows:

4.2.1 Basic Features

Basic features refer to the Bag-Of-Words features. Similar to document classification, we use the appearance of terms in each sentence as the first set of features. However, unlike document classification, traditional **stopwords** in IR play an important role in route parts. For example, "take", "onto" and "at" are essential in instructions.

4.2.2 Surficial Features

Surficial features refer to the visual features observable directly from the sentence, for example, whether a sentence has only one word or consists of characters other than letters and digits, whether all the words are capitalized. We chose this set of features in order to characterize the route parts expressed in single words or phrases. For example, destinations frequently appear as the name of a business or

Algorithm 1 Sentence Delimitation in HTML

Input: An HTML document *doc*, a tags-to-skip list *skipList*, a tags-to-concatenate list *concatList*
Output: A list of sentences *sList*
Procedure:
1: $sList \leftarrow \emptyset$; $tList \leftarrow \emptyset$; String $t \leftarrow \emptyset$; $flag \leftarrow true$
2: parse *doc* into DOM tree *dTree*;
3: **repeat**
4:     let n be the next node to visit during depth-first traversal of *dTree*;
5:     **if** n is Text Node **then**
6:       append n's text to t;
7:     **else if** n is Tag Node **then**
8:       **if** n is in *skipList* **then**
9:         skip the subtree rooted at n;
10:       **else if** n is in *concatList* **then**
11:         **if** $flag ==$ false **then**
12:           $t \leftarrow \emptyset$;
13:         **end if**
14:         $flag \leftarrow true$;
15:       **else**
16:         $flag \leftarrow false$; put t into *tList*; $t \leftarrow \emptyset$;
17:       **end if**
18:     **end if**
19: **until** all nodes in *dTree* has been visited or skipped;
20: **for** each text piece t in *tList* **do**
21:     parse t into sentences and put them into *sList*;
22: **end for**
23: **return** *sList*;

---

an organization and all the words in the name are capitalized; sentences having no letters or digits in it are frequently labelled as "other".

4.2.3 Visual Features

We extracted a set of HTML visual features such as whether a sentence is a title, a link or a heading, etc. This is based on our observation that HTML authors usually use different visual features for different route parts. For example, titles of HTML documents usually contain the destination; destinations and origins are usually in Headings; links in HTML are usually irrelevant to route parts.

4.2.4 Domain-specific Features

One set of domain-specific features are **language patterns**. We identified a set of frequent patterns in direction descriptions. Such patterns include highway names and particular verb phrases, such as "turn left to ...", "merge onto ..." and "proceed ... miles...". A rule-based approach using string pattern matching is applied to generate this set of features. A set of rules is predefined and carefully examined. Table 2 gives examples of regular expressions and language patterns in the text (HighwayPS is a pattern string for matching highway names). We designed 25 regular expressions to extract frequent language patterns for instructions, 2 for destinations, 1 for origin and 2 for other. We tried to make the set of regular expressions as compact as possible. So if multiple phrases can be put into one regular expression, we do so to reduce the size of the rule set. Thus the number of matched phrases and word combinations is much larger than the number of regular expression. Another set

	Key Words	Num of Docs
1	direction, turn, go, mile	986
2	direction, turn, left, right, exit	775
3	direction, turn, mile, take, exit	588

Table 3: Sample search key words and number of Docs obtained

of domain-specific features are nouns and noun phrases that can be encoded in a **dictionary**. We created a dictionary of frequent nouns and noun phrases referring to a place or a location, such as "hotel", "restaurant", "campus", etc. The dictionary has 110 entries. We build this dictionary based on our observation that the entries are usually the agencies hosting the driving direction web pages and these nouns or noun phrases frequently appear in the destinations.

4.2.5 Other Features

In addition to the above feature sets, we also included a set of "Window Features". Window features capture the characteristics of surrounding text of a sentence. Window features are extracted after the surficial and language pattern features are extracted. It checks the existence of one or a set of specified features in the window surrounding the current sentence. For example, whether there is an "INST" feature in the sentence before or after the current sentence; whether the previous and following 2 sentences all have a certain feature, etc.

5. EXPERIMENT RESULTS

In this section, we first describe how we build our data set. Then we evaluate the performance of sentence delimitation and classification algorithms.

5.1 Data Set and Document Classification

A set of over 11,000 web pages containing route directions were identified using the search results of the Yahoo![4] 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 they are typically present within documents containing route directions. Manual examination shows 96% of these documents contain route directions. A randomly selected subset of 10,000 web pages from the random sampling of the web using the method proposed by M. R. Henzinger, et al. [5] is used as the negative examples. Table 3 shows some examples of search queries and number of unique documents obtained from the returned result pages. We trained a Maximum Entropy classifier for a binary document classification task. It yields an average of over 98% accuracy over 5 rounds of test.

5.2 Sentence Extraction Evaluation

We compare the effectiveness of our proposed hybrid sentence delimitation algorithm (HYD) with two other approaches: the plain-text-based (PTB) method, which converts an HTML into plain text format and then used natural language knowledge to segment sentences, and the HTML-tag-based (HTB) method, which parses an HTML document into a DOM tree and extracts the text nodes as sentences.

The three algorithms are applied to the same set of HTML documents containing 403 human-identified sentences. For

---

[4]www.search.yahoo.com

Feature Name	Regular Expressions	Example
INST 1	.*follow \\s \\d{1,5}(?: \\. \\d{1,5})? \\smile(s)?.*	"... follow 3.4 miles..."
INST 2	".*exit \\s+(?:at \\s+)?" + HighwayPS + ".*"	"...exit at PA Route 23..."
DESTINATION	"\\s*(?:driving)?\\s*(direction|directions)\\s+to\\s+\\w{2,}.*"	"driving directions to IST"

Table 2: Sample Regular Expressions to extract domain-specific features

	HYD	PTB	HTB
correctly-extracted	**391**	152	226
over-segmented	7	5	32
under-segmented	5	246	145
accuracy	**97.02%**	37.72%	56.08%

Table 4: Sentence extraction results

each algorithm, we counted the number of sentences of three types: 1) correctly extracted, 2) over-segmented sentences and 3) under-segmented. Correctly extracted sentences are the human-identified sentence. If one human-identified sentence is broken into several pieces by the algorithm, we count one over-segmented sentence. If n human-identified sentences are concatenated, we count n under-segmented sentences. Table 4 shows the details.

5.3 Cross Validation Method

In order to explore the effectiveness of models which consider the dependencies between sentences, the ordering between sentences in one document should be preserved. Therefore, we shuffle the order of documents, instead of the sentences, so that the ordering of sentences within each document can be preserved. Then the documents are divided into k equal-sized groups. In each training-testing cycle, one group is used as testing set and the remaining $k-1$ groups are used as training set.

5.4 Sentence Classification

We evaluated four models: Naive Bayes (NB), Maximum Entropy (MaxEnt), CRF and MEMM. For CRF and MEMM, we changed the value of initial Gaussian variance to be 1.0, 5.0 and 0.5. In order to evaluate the impact of different feature sets, we divided the features into 5 groups: Bag-Of-Words(B), Language Patterns and surficial features matched by regular expressions(R), Window features(W), HTML visual features(H) and Dictionary(D). We add the features one by one, i.e. B, BR, BRW, BRWH and BRWHD. Besides, we tested the performance of these features without traditional IR stopwords. Each model is applied on 10 different feature sets. The 10-fold cross validation technique described above is applied on 100 HTML documents containing over $10,000$ human tagged sentences. A total of $9,880$ sets of experiments were conducted (Window feature is not used for NB and MaxEnt). Due to space limits, only part of the experimental results are shown.

Figure 1(a) shows the sentence classification accuracy of different models on the full feature set (BRWHD) with stopwords. CRFs and MEMMs outperform NB and MaxEnt. The reason is that some sentences without strong features of route parts can be inferred by the states of adjacent sentences by CRFs and MEMMs, but are hard for NB and MaxEnt to recognize. For example, an instruction "east 7.4 mi" was not recognized by NB or MaxEnt, but was recognized correctly by CRFs and MEMMs because its previous
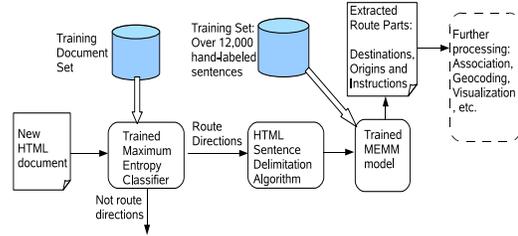

Figure 3: System Architecture of First Model of GeoCAM

and following sentences are both instructions.

The effects of different feature sets are shown in Figure 1(b). We start from Bag-Of-Words (B) features only, then we add in language patterns, denoted by BH; then window features and so on. As more features are added, the performance steadily improves. We notice that among all models, language patterns give the largest improvement. Figure 1(c) shows the importance of using traditional IR stopwords in sentence classification. They give a significant improvement in route part sentence classification because most route parts, especially instructions contain many characteristic stopwords. This confirms that stopwords are domain-dependent. Figure 2 shows the precision, recall and F1 score of each model for each class.

As can be noticed in Figure 2, although the classification accuracies for Instruction, Other and Origin are high and reasonable, the recognition of destination is a hard problem for all the four models. This is because: 1) the position at which a destination appears in the text is less regular compared to the other 3 classes;2) there lack a set of features that best characterize destinations. Although we identified some language patterns for destinations, they are frequently described in only business names which don't have very obvious language patterns; 3) destinations are usually very short, thus making bag-of-words features perform poorly in the recognition. A potential solution is to use geography databases to search for business names that match the business name in the text.

6. GEOCAM SYSTEM

The research reported in this paper is part of our GeoCAM project. The first module of the system allows users to upload an HTML document. Then the system classifies the document as either "Direction" or "Non-direction" using a trained Maximum Entropy classifier. The system then extracts sentences from the HTML and uses a learned MEMM sentence classifier to assign one of the following labels to each sentence: "Destination", "Origin", "Instruction" and "Other". Based on the classification result, the sentences in the HTML document are highlighted with different colors. Figure 3 shows the architecture of the first module.
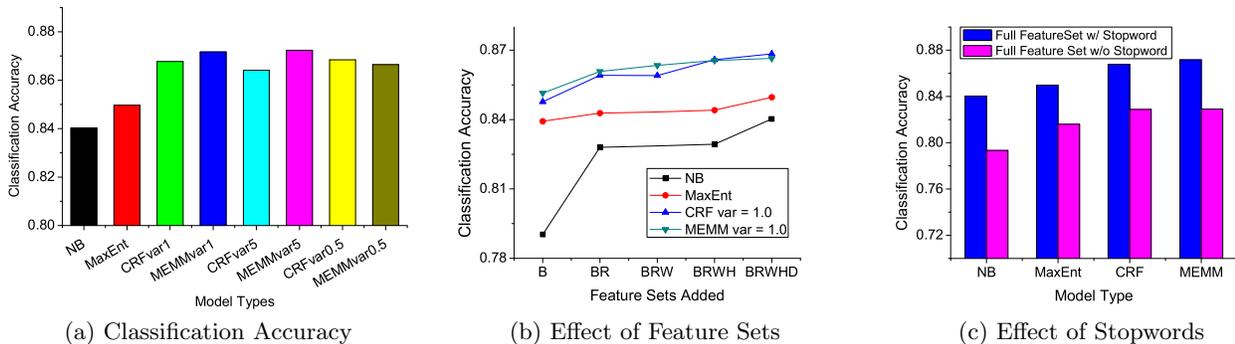
(a) Classification Accuracy   (b) Effect of Feature Sets   (c) Effect of Stopwords

Figure 1: Experimental Results

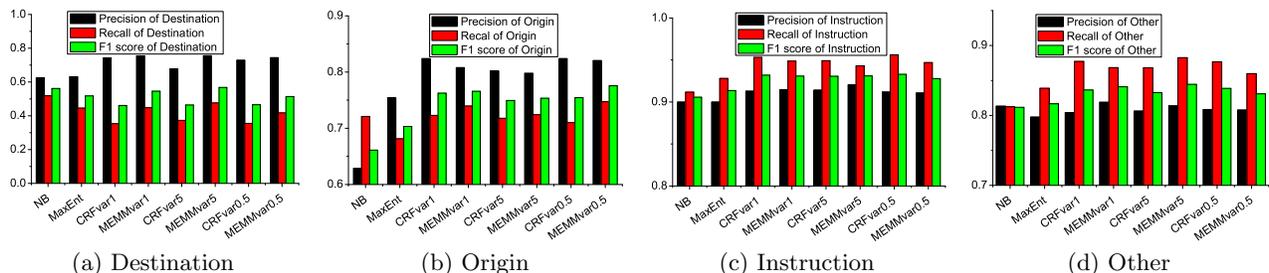(a) Destination   (b) Origin   (c) Instruction   (d) Other

Figure 2: Detailed Analysis of Each Class

7. CONCLUSIONS AND FUTURE WORK

In this paper, we showed the first work toward automatic extraction of route parts in HTML documents. We studied the problem of sentence extraction from HTML documents and analyzed the inter-dependency of sentences within a document. Our proposed sentence extraction algorithm provides a good solution to the HTML sentence extraction problem. Besides, we showed that sequence labelling algorithms such as CRFs and MEMMs outperform other models based on independence assumptions. Moreover, we explored various sets of features and studied the effects of them in sentence classification. Given a new sentence classification task, the feature sets introduced in this paper, together with other feature sets in previous work, are potential useful feature sets and worth investigating. Although some of the features mentioned in this paper are dedicated to route directions, such as the regular expressions, they can be generalized to other sentence classification tasks if changes are made accordingly. We identified the problem in the poor performance in the recognition of destination and are going to explore the possibility to use a geography database to improve the performance. We also introduced the GeoCAM system which demonstrates the initial effort of automatic extraction of motion descriptions in text. Multiple destinations and origins may appear in one document and 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.

8. REFERENCES

[1] A. L. Berger, S. A. D. pietra, and V. J. D. pietra. A maximum entropy approach to natural labguage processing. In *Computational Linguistics*, 1996.

[2] A. Borthwick. A maximum entropy approach to named entity recognition. In *Ph.D. thesis, New York University*, 1999.

[3] D. Freitag and A. McCallum. Information extraction using hmms and shrinkage. In *AAAI Workshop on Machine Learning for Information Extraction*, 1999.

[4] B. Hachey and C. Grover. Sequence modelling for sentence classification in a legal summarisation system. In *Proceedings of 2005 ACM Symposium on Applied Computing*, 2005.

[5] M. Henzinger, A. Heydon, M. Mitzenmacher, and M. Najork. On near-uniform url sampling. In *WWW*, May 1999.

[6] N. Jindal and B. Liu. Identifying comparative sentences in text documents. In *Proceedings of SIGIR*, pages 244–251, 2006.

[7] A. Khoo, Y. Marom, and D. Albrecht. Experiments with sentence classification. In *Proceedings of ALTW*, 2006.

[8] A. Klippel and D. R. Montello. Linguistic and nonlinguistic turn direction concepts. In *In Preceedings of COSIT*, 2007.

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

[10] D. D. Lewis. Naive (bayes) at forty: The independence assumption in information retrieval. In *Machine Learning: ECML 98*, pages 4 – 15, 1998.

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

[12] F. Peng, F. Feng, and A. McCallum. Chinese segmentation and new word detection using conditional random fields. In *Proceedings of COLING*, pages 562 – 568, 2004.

[13] L. R. Rabiner. A tutorial on hidden markov models and slelcted applications in speech recognition. In *IEEE*, 1989.

[14] A. Ratnaparkhi. A maximum entropy part-of-speech tagger. In *EMNLP*, 1996.

[15] B. Sun, P. Mitra, and C. L. Giles. Mining, indexing and searching for textual chemical molecule information on the web. In *Proceedings of WWW*, 2008.

[16] W. Sunayama, A. Iyama, and M. Yashida. Html text segmentation for web page summarization by a key sentence extraction method. In *Systems and Computers in Japan*, 2006.

[17] L. Zhou, M. Ticrea, and E. Hovy. Multi-document biography summarization. In *Proceedings of EMNLP*, 2004.