Named Entity Recognition and Classification

1. Choosing project

We will implement a project that proposes to implement the identification of proper names in texts, and classification into a set of pre-defined categories of interest as:

- Person names (names of people)
- Organization names (companies, government organizations, committees, etc.)
- Location names (cities, countries etc)

Our motivation is to continue with the project after this semester and implement search engines, question answering systems that can be enhanced by allowing searches and questions for particular persons, companies or locations.

Our approach is to take the following algorithms and to see which one is capable to ensure the best accuracy:

- Hidden Markov Model (HMM)
- Decision Trees (DT)
- Maximum Entropy Models (ME)
- Support Vector Machines (SVM)
- Conditional Random Fields (CRF)

We want to search all the papers suitable for an application using english and until now, the main paper we found is:

Named Entity Recognition: A Literature Survey by Rahul Sharnagat [1]

We believe that starting with an overview of the papers that appeared on this subject is the best approach to continue with research.

[1] http://www.cfilt.iitb.ac.in/resources/surveys/rahul-ner-survey.pdf

2. State of the Art

2.1. Introduction

For this milestone we had to study the state of the art for the chosen topic NERC.

A good proportion of work in NERC research is devoted to the study of English but a possibly larger proportion addresses language independence and multilingualism problems

Textual genre and domain factor are one of the most common issue for a general language independent solution. T. Poibeau and Kosseim (2001), tested some systems on both the MUC-6 collection composed of newswire texts, and on a proprietary corpus made of manual translations of phone conversations and technical emails. They report a drop in performance for every system (some 20% to 40% of precision and recall) (David Nadeau, Satoshi Sekine)

Supervised learning is the dominant technique for addressing NERC problem.
It includes :

- Hidden Markov Models (HMM) (D. Bikel et al. 1997),
- Decision Trees (S. Sekine 1998),
- Maximum Entropy Models (ME) (A. Borthwick 1998),
- Support Vector Machines (SVM) (M. Asahara & Matsumoto 2003),
- Conditional Random Fields (CRF) (A. McCallum & Li 2003)

We decided to study the following topics related to our main project:

- Alexandru Ciobanu - Maximum Entropy
- Dan Bizdadea - Support Vector Machines
- Florentina Bratiloveanu - Neural Networks
- Mihaela Sorostinean - Decision Trees

Following, each of us will describe the state of the art of our chosen topic.

2.1. Maximum Entropy

I have read the following papers :

- ”Maximum entropy models for named entity recognition." By Bender, Oliver, Franz Josef Och, and Hermann Ney
- "Language independent NER using a maximum entropy tagger." Curran, James R., and Stephen Clark.
- "Named entity recognition: a maximum entropy approach using global information." Chieu, Hai Leong, and Hwee Tou Ng

Each paper uses the maximum entropy model but the features used for tagging are different and use different strategies.

The first paper uses features such as Capitalization, Digits and Numbers, and Prefixes and suffixes for tagging. It also suggest a method for feature selection in which features are selected if the features are observed at least k times, where k is the predefined threshold.

After tagging the highest probability sequence is selected using a Viterbi search. Some words can also appear only a couple of times as cannot be tagged. In this case the model suggest the usage of a lexical feature, that taggs these words as <unknown>.

The second article presents a model for implementing a language independent NER. The basic algorithm is similar to the one presented in the first article, the differenced being the features and the usage of gazetteers. The article also provides sample features such as: checking if the word contains periods, punctuation, is a Roman numeral of is an initial. The main difference from the previous model is the usage of gazetteers. These are useful for language independency as they can hold information about a particular word in many languages. E.g. the name John is Ion in romanian and Joshua in Hebrew.

The third model has a different approach during the preprocessing phase. The authors start by creating lists from the training data. Such list can be: frequent words list, useful unigrams, useful word suffixes or so on. After the preprocessing stage features are used for word tagging. The likability for a tag to be attributed to a word is based on the content of the previously created lists. The features used include: acronyms, unigrams or bigrams.

2.2. Support Vector Machines

Support Vector Machines (SVMs) based NER system was proposed by Yamada et al. for Japanese

Other NER systems have been designed by :

- Koichi Takeuchi & Nigel Collier
- Masayuki Asahara & Yuji Matsumoto
- Asif Ekbal & Sivaji Bandyopadhyay

Their NER system includes two main phases: training and classification. The training has been carried out by YamCha 3 toolkit, an SVM based tool for detecting classes in documents and formulating the NER task as a sequential labeling problem. For classification, they have used TinySVM-0.07 4 classifier that seems to be the best optimized among publicly available SVM toolkits.
None of these systems have been developed for the English language and given the high difference of precision / recall scores between different languages and text-domains, I can't consider their results as a measure for our work.

I will consider as a benchmark the Stanford NLP Conll 2003 English news testa data results as follows for 3 of their classifiers :
Pure CMM :
Precision: 91.37%
Recall : 91.22%
F1 : 91.29%
Postprocessed CMM :
Precision : 92.15%
Recall : 92.39%
F1 : 92.27%

CRF (with distsim) :

Precision : 93.28%
Recall: 92.71%
F1 : 92.99%

2.3. Neural Networks and Word2vec

Because, I couldn't find some papers that specifically implement a combination between neural networks and word2vec, I'll try to expose my idea.

Word2vec is a two-layer neural net that receives as input a corpus and outputs feature vectors for words in that corpus. Basically, it converts text to numbers and it's a good algorithm in finding similarities between words(**e.g** "man" is to "boy" that "woman" is to "girl"). One of the papers that mention word2vec and NER is "Adapting *word2vec* to NER". The goal is to add new information to a classifier.

- The corpus used is Reuters Corpus Volume 1
- The algorithm used to classify is Linear SVC
- The best accuracy they achieved was 83.52%

For backpropagation algorithms I found different types of networks:

1. Twinet Recurrent Networks for Named Entity Recognition with a F1-score of 86.20%. They say that every word in a sentence can make contributions in deciding whether a chunk is a named entity or not.
2. Boosting Named Entity Recognition with Neural Character Embeddings is another paper that introduces deep learning and achieves a F1-score of 82.21%.

My intuition is that we can use word2vec in neural networks/deep neural nets for various reasons: to avoid overfitting, to extend the dataset and to give a more flexible representation for the words, such that networks can understand the input.

Even if it's not a paper, I will try to use some conv net proposed by Eric Yuan where he uses a small dataset about 1000 sentences of news where he has a 96.9% accuracy per words. He tried to train using word2vec and convnets and I will try to adapt the convnet to the dataset that I will be used and its length.

- The number of categories is 9
- The model used for convnet is:

Convolutional(1) –> Pooling(1) –>Convolutional(2) –> Pooling(2) –> Fully Connected –> Softmax

2.4 Conditional Random Fields

Conditional Random Fields (CRF) - undirected graphical models
-used to compute conditional probability of values for output nodes given assigned values for the input nodes

There are a number of works that used CRFs for information extraction and words recognition:

- In a work presented by McCallum et. al (McCallum, Andrew, and Wei Li. "Early results for named entity recognition with conditional random fields, feature induction and web-enhanced lexicons." *Proceedings of the seventh conference on Natural language learning at HLT-NAACL 2003-Volume 4*. Association for Computational Linguistics, 2003.) a method is presented for feature induction for CRFs. The paper describes a method called *WebListings* used to obtain seeds for the lexicons from the labeled data and then use the Web to augment those lexicons. The authors reported an overall F1 of 84.04% for English and 68.11% for German.

- In another work Sarawagi et. al (Sarawagi, Sunita, and William W. Cohen. "Semi-markov conditional random fields for information extraction." *Advances in neural information processing systems*. 2004.) used semi-Markov CRFs (conditional trained version of semi-Markov chains) in experiments on five named entity recognition problems. According to the results reported by the authors, semi-CRFs generally outperforms conventional CRFs.

For the next step of the project, after a thorough analysis of the results reported in the previous works I will try to make an implementation and also suggest some possible improvements.