Project Progress Report

CS410 Text Information Systems
University of Illinois at Urbana Champaign
by Andrey Nedilko

Before starting the project, I decided to do a brief research of the current text classification/categorization advancements in order to understand how I can create a meaningful application or use already available resource more effectively.

I have reviewed the following APIs and related software systems:

1) **Cogito Intelligence API** (includes semantic analysis)
Key Features

- Content Categorization with specialized classification for intelligence taxonomy, crime taxonomy, terrorism taxonomy, cyber crime taxonomy, geographic taxonomy
- Entity, Time References (alpha version), Facts and Relationships Extraction, over 50 domain entities (world leaders, terrorist organizations, organized crime, unconventional weapons, geographic locations, etc.)
- Writeprint for stylometric characterization and authorship assessment
- Emotions detection (anger, fear, disgust, sadness, happiness, surprise, etc.) related to the entities within the text
- Social Metadata Disambiguation (e.g. #bringbackourgirls transforms into [bring back our girls])
- Semantic Reasoning for surfacing “hidden” entities
- Writeprint for stylometric characterization and authorship assessment

This **free web service includes all the 16 channels** available: full categorization, intelligence categorization, crime & offense categorization, terrorism categorization, cyber-crime categorization, geographic categorization, emotions, people text mining, organization text mining, place text mining, domain entities text mining, fact mining, summary, writeprint and semantic relations between entities.
Web service calls are limited to **one every 100 seconds**. The free option is limited to **two months of service**.

2) https://en.wikipedia.org/wiki/Weka_(machine_learning)
Weka - a collection of visualization tools and algorithms for data analysis and

predictive modeling, together with graphical user interfaces for easy access to these functions

3) MeaningCloud's Text Classification System
https://www.meaningcloud.com/developer/text-classification
A trainable model. Text Classification is MeaningCloud's solution for automated document classification. It assigns one or more categories to a text, using standard domain-specific taxonomies (e.g., IPTC. IAB, ICD-10) or user-defined categories. The algorithm combines statistical document classification with rule-based filtering, which allows to obtain a high degree of accuracy and flexibility in a wide range of environments.
Differentiators:

- Statistical classifiers provide a means to use example documents to define each category.
- In turn, rule base classifiers helps to fine-tune the classification and correct the output of statistical classifiers.
- Its powerful rule based classification language is also useful to bootstrap a categorization when no examples are available.
- Features predefined, standard taxonomies: IPTC, IAB, ICD-10, Eurovoc.
- User can create his/her own categories and classification models.

4) Open Calais
http://www.opencalais.com/about-open-calais/ (does extract some relevant topics, but several)
Open Calais (has a free API) is a free service currently accessible via a public website (opencalais.com) and will also be available via a Thomson Reuters sponsored public website called PermID.org. This free service provides document tagging using an extensive set of fields such as Company, Person, Geography, Industry Classifications, Topics, Social Tags, Facts, and Events. The service is hosted by Thomson Reuters and allows users to upload up to 5,000 documents a day.
Open Calais is a sophisticated Thomson Reuters web service that attaches intelligent metadata-tags to your unstructured content, enabling powerful text analytics. The Open Calais natural language processing engine automatically analyzes and tags your input files in such a way that your consuming application can both easily pinpoint relevant data, and effectively leverage the invaluable intelligence and insights contained within the text.

Open Calais analyzes the semantic content of your input files using a combination of statistical, machine-learning, and custom pattern-based methods. Developed by the Text Metadata Services (TMS) group at Thomson Reuters, Open Calais outputs highly accurate and detailed metadata. Open Calais also maps your metadata-tags to Thomson Reuters unique IDs. This supports disambiguation (and linking) of data across all documents processed by Open Calais, and also offers you the opportunity to further enrich your data with related information from the Thomson Reuters datasets.

Open Calais automatically analyzes your input text and performs the following processes:

• Named Entity and Relationship Recognition – Open Calais identifies and tags mentions (text strings) of things like companies, people, deals, geographical locations, industries, physical assets, organizations, products, events, etc., based on a list of predefined metadata types.

• Aboutness Tagging – Open Calais assigns social, topic, and industry tags that describe what the input document is about as a whole.

I also found the following sources quite useful or interesting:

1) http://textblob.readthedocs.io/en/dev/classifiers.html#
Tutorial: Building a Text Classification System with TextBlob

2) http://stevenloria.com/how-to-build-a-text-classification-system-with-python-and-textblob/
Tutorial: Simple Text Classification with Python and TextBlob

3) https://blog.statsbot.co/text-classifier-algorithms-in-machine-learning-acc115293278

4) https://www.clips.uantwerpen.be/pages/pattern-en

5) https://towardsdatascience.com/

Currently, I am working on the theory of contextual representation of words (for automated word sense disambiguation) and word representation by word context vectors (word embeddings)

I am relatively new to this field which I actually like very much, and therefore I feel like I have absolutely no difficulty in understanding the theory, but I may lack some practical skills here or there. That is why I would really appreciate any advice that you can give me or if you could please send me on a more straightforward path in the right direction. So far, I came up with the following questions, but any advice is welcome:

- As far as I realize, section *Datasets and Dataset Views* in the classification tutorial https://meta-toolkit.org/classify-tutorial.html is related to C++. Are there any materials describing how I can effectively engage metapy in the text classification/categorization process?
- If I am planning to have a training set of documents in each category, what format should the documents be in? What kind of preformatting / preprocessing should I do before trying to create an index?