Language Independent Entity Recognition and Typing

Jayachandu Bandlamudi, Neil Patel

August 10, 2017

Abstract

Entity recognition and typing is an important study due to the many forms of text collections that are emerging, such as mobile applications and social media. Most text data is unstructured and does not provide much meaning or knowledge. Entity recognition and typing will structure the data and provide more meaning about the text data. In this paper, we will focus on retrieving quality entity phrases, and with the use of a phrase-based entity recognition framework, ClusType [1] that propagate the entity type information through co-occurring relational phrases and their surface name's type information. This paper focuses on improvements to the input data for the ClusType propagation. The quality of the seed mentions and entity phrases will determine the quality and accuracy of the propagation algorithm and the effectiveness of the entity typing. In our work we used Autoencoder neural network to look for anomalies in the seed entity mentions based on word embeddings. Seed entity mentions that are anomalies will be removed and not used to guide the typing. Also, during entity detection collocation based filtering is performed such that entity phrases that are not high quality will be excluded from typing. Details about the framework will be discussed later.

1 Introduction

Named Entity Recognition has been studied by many researchers in Text Mining and Natural Language Processing. It is considered as the process of Information Extraction from large unstructured text corpora in order to obtain structured information. For example, News articles are rich source of text and many times the whole article is of little use. An article can be better summarized with the extracted entity mentions such as PERSON, ORGANIZATION,LOCATION to interpret the high level context.

Definition: Entity is a word or phrase which is a distinguishable type item that exists in a group of different typed items and these types refer to predefined categories such as PERSON, LOCATION, ORGANIZATION etc., and the process of identifying the potential entities and classifying them into predefined types is called Named Entity Recognition(NER).

Type information can be collected from knowledge bases (KB), like Freebase [2] and DBpedia [3], see figure 1. Even with very massive databases of entity mentions, the collections do not cover a majority of entities. Weak supervision and distant supervision are two different methods to extend the typings to entities beyond the available entities in KBs. This paper will focus on a distant learning method for entity recognition. In distant supervision, the entities will need to be collected from the text corpus, and those entities will be mapped to the KB and typed. The

Figure 1: Knowledge Base and Entity Linking

remaining unmapped entities are candidate entities, which will require a propagation from the mapped entities.

2 Background and Problem

Distant learning is an efficient way to type entity phrases. The knowledge base contains many entity phrases and their corresponding types, but some are missing and some entity phrases are easier to type more than others. For example, "Washington" is a PERSON, LOCATION or ORGANIZATION. For entities that have multiple types, it will be hard to type without the context of the sentence. Therefore, distant learning is a great method to obtain initial typing of entity mentions, but some entity mentions may have erroneous typing. Since this initial typing is really important for the rest of the propagation algorithm, we are going to focus on removing the erroneous typing of some entity mentions.

The knowledge base also represents entity phrases that are common and used widely in the English language. The entity phrases that are a jumble of a few entity phrase will not appear in the KB. It is important to detect quality entity phrases for the typing to be accurate and precise. Entity phrases that have poor quality are also hard to type because the words that compose the entity phrase may have different types. It will be important to separate these entity phrase to get better KB results and propagation results.

This paper studies distant learning and improving the effectiveness of the ClusType framework, which types entities to predefined categories for a domain-specific corpus. ClusType uses POS tagging on the corpus to retrieve candidate entity mentions and relational phrases. It uses those phrases for typing and building a graph that will propagate the labeled type to the non-typed candidate entity mentions. Therefore, the input is a very important part of the framework, since the propagation is effective and accurate if the initially typed entity mentions are typed correctly. It is also important to find quality entity phrases, because poor entity phrase will not have an entry in the KB. Poor entity phrases may also get typed after the propagation, but due to the poor quality of the phrase, the typing will be erroneous and the results may propagate and cause more erroneous typings.

In this study will target the input of the ClusType framework i.e; detected entity phrases and seed mentions. The candidate mentions and the typing should have good quality phrases and the phrases should be typed correctly from the KB. The challenges that we are focusing on is targeted towards improving the quality of the typing, which is obtained from the KB, and the entity phrases that are detected by the ClusType framework.

Phrases that have words that do not belong together may represent different types, like "Yankees in New York" represents an organization and a location. It is important to break this phrase into its two appropriate entity phrases. For this challenge, we follow the same method as ClusType, phrase mining on a POS-tagged corpus. After this step, the corpus will be tagged with candidate entity mentions (entity phrases). We will use a collocation measure to remove phrase that do not have a high score. Collocation of words is a conventional way of constructing a phrase. In the previous example, if the phrase was "New York Yankees", these words can be standalone but this is a common way to express the baseball organization. Collocation measure for this phrase will be high versus the previous phrase.

Another input to the ClusType Framework are the seed entity mentions. These are the candidate entity mentions that are typed via the KB. Depending on the corpus' domain, the knowledge base can type some percent of the candidate entity mentions, and the rest of the candidate entity mentions will get type through the propagation algorithm. It is important to detect the anomalies in the seed entity mentions. If there are entities with erroneous typings, the types could propagate to other entities. For this challenge, we used Autoencoder based anomaly detection using word embedding. Entity mentions need to be converted to vectors of real numbers in order to relate the words to each other. Words that are located closer in the vector space share common contexts in the corpus. Therefore, these words should be related and their types should most likely be the same. In this paper, we use Word2Vec [4], which takes a text corpus and converts the words and phrases to vector space. A previous extension to the ClusType framework, Partial-Label Embedding [5] (PLE), also utilized low-dimensional space. PLE used low-dimensional space with fine grained entity typing. We use a similar idea in this paper. If the seed entity phrases cluster in one area of low-dimensional space, that area should represent the same type of words. For example, the "New York Yankees" and the "Boston Red Sox" should be located in close proximity to one

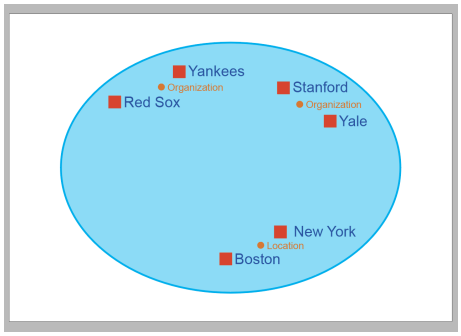

Figure 2: Word Embedding

another, see figure 2. Therefore, we can hypothesize that these entity phrases carry the same typing. In this case, the two entities should be organizations.

Integrating these methods prior to the ClusType algorithm allows the propagation algorithm to use quality entity phrase, which are accurately typed. The major contributions of this paper are as follows:

1. In addition to an already robust framework in ClusType. The new addition will filter entity phrases and seed entity phrase that are incorrectly used in the input of the propagation algorithm.

2. A filtering process is used to clean seed entity mentions using Autoencoder based anomaly detection.

3. Entity phrases having poor collocation are not considered for typing, and this paper addresses the collocation of all entity phrases that have more than one word.

3 Collocation Measures

As discussed, Clustype framework detects entity mentions based on the grammatical rules and POS tagging techniques such as Consecutive Nouns, Consecutive Capitals. And this approach for entity mention detection is simple and often entity mentions are of low quality such as "Okay Texas" which is expected in case of bi-grams, trigrams, etc. In our work, we incorporated collocation based association measures to obtain high quality entity mentions. At first we generate entity mentions using the Clustype methodology, so that we don't miss any mentions and then retain only high quality entity mentions based on collocation score.

Lexical association measures/collocation measures are mathematical formulas determining the strength of association between two or more words based on their occurrences and co-occurrences in a text corpus. There exists many collocation measures while assessing the quality of an entity mention [8] and we experimented with two such methods PMI (pointwise mutual information) and t-score based method for significance of an entity mention.

3.1 PMI

Point wise mutual information (PMI) is a measure of association used to assess the quality of an entity mention. Entity mentions having PMI score above threshold (>5) are considered high quality and often these collocations are found in text corpus. And the mentions with low PMI score are not considered as entity mentions and typing is not performed.

$$pmi(w1;w2) \equiv log\frac{p(w1,w2)}{p(w1)p(w2)} = log\frac{p(w1|w2)}{p(w1)} = log\frac{p(w2|w1)}{p(w2)} \quad (1)$$

Where w is the n-gram entity mention frequency in the corpus and w1, w2, etc. are unigram frequencies in the corpus.

It can take positive or negative values, but is zero if w1 and w2 are independent. Note that even though PMI may be negative or positive, its expected outcome over all joint events (MI) is positive. PMI maximizes when w1 and w2 are perfectly associated

3.2 t-score

Collocation in a text corpus can be measured by Student's t-test. In a bi-gram phrase, equation 2 represents the unconditional probability of the first word and 3 is the probability of the second word. The t-score can be seen in equation 4.

$$P(w_1) = ws_1/N \tag{2}$$

$$P(w_2) = ws_2/N \tag{3}$$

$$t = \frac{\bar{x} - \mu}{\sqrt{\frac{s^2}{N}}} \tag{4}$$

$\bar{x}$ is the mean of the occurrence of words one and two. μ is the probability of words one and two when assuming that they appear independently in the corpus. s^2 is the sample variance.

Considering the above discussed method for significant collocation we used p-value of 0.01 for the two-tailed z-score test. Under Alternate Hypothesis (Ha: Collocation is significant if P(-2.576 <= X <= 2.576) = 0.99 and NULL hypothesis (H0: words in collocation appear independently in corpus).

4 Anomaly Detection

As discussed earlier, seed entity mentions guides the typing of entity mentions and often seeds obtained from KB are noisy. Seed mentions from KB can be filtered based on the quality score from KBs but this approach does not guarantee the novelty of seed mentions. In our study, we explored the option of detecting anomalous seed entity mentions and these anomalies are discarded from positive examples for better Clustype training.

Anomaly detection is based on a statistical detection of outliers. The detection is based on points that fit a probability distribution that fits data model [6]. If a data point does not fall within a selected threshold based on the probability of that data point in that data model, the point is considered an anomaly. Like a probability based data point, a clustering algorithm can evaluate data points by the distance from the clusters. If the distance is beyond the selected threshold for the problem, the data point is considered an anomaly. This is also similar to distance based measures like KNN. Anomalies will have extreme KNN distances and they will not belong to any groupings. Certain data points will require a new error or anomaly score for measurement. For this method, we can use Autoencoder [7]. Using the difference between the k-most significant principal components and the original data point, we can label high differences as anomalies.

There are several methods for anomaly detection, but we used state of the art neural network based Autoencoders [7] framework and the feature set for this unsupervised learning task are word embeddings obtained by word2vec model [4].

First step in this seed mentions anomaly detection task is generating feature set for learning, as we know our seed mentions are tokens of words and frequency/count based methods for feature generation will yield sparse vectors. Therefore, we used word embedding based feature representation in which each seed mention can be represented by a dense numeric vector of fixed size.

A word2vec model will be trained on raw corpus, to represent embeddings for the unique seed mentions irrespective of their type.The idea of word embeddings is intuitive as entity mentions of same type are in the proximity in embedded space compared to a different type. And these word embeddings also contain latent features such as relational phrases pertaining to entity mentions.

4.1 Word2Vec

Word-embeddings is a feature learning method in which words or phrases in the vocabulary are represented by low dimensional real number vectors. Feature vectors learned from word-embeddings are powerful and represent the context of words in the text. For example, by the vector representation of vec("Madrid") - vec("Spain") + vec("France") is closer to vec("Paris") than other words in corpus, which means the Country-Capital relation is embedded in the feature vectors.

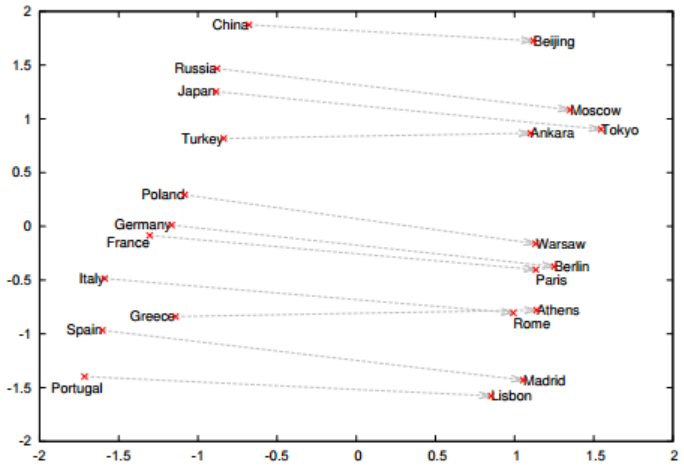

Figure 3: word2vec embeddings for country-capital relation

In Figure 3, we observe that the distances between pairs of country and its capital are almost equal. Word2vec learns embeddings using supervised neural network method and either of the architectures continuous bag-of-words (CBOW) or continuous skip-gram.

4.2 Autoencoder

Autoencoder neural network is an unsupervised learning algorithm that applies back-propagation. It will reconstruct the target values to be equal to the inputs. Equation (5) represents the encoder and equation (6) represents the decoder.

$$h = \sigma(Wxhx + bxh) \tag{5}$$

$$z = \sigma(Whxh + bhx) \tag{6}$$

$$kx - zk \tag{7}$$

W and b is the weight and bias.σ is a nonlinear transformation function. The reconstruction error is the difference between the input vector and the reconstruction vector (7). The goal of the Autoencoder is to minimize the reconstruction error. Anomaly detection is a semi-supervised learning method. It uses the reconstruction error as the anomaly score. The data points that yield a high reconstruction error is an anomaly.

5 Framework

5.1 Candidate Generation

The candidate generation step involves detecting entity phrase and relation phrase in the text corpus. Each relation phrase, like “of”, “in”, “at”, etc., could have zero to two entity phrase, and it is a way of linking those entities to each other. In this step, the entity phrases are marked with their corresponding relation phrase. The candidate generation uses methods like POS tagging to obtain nouns as entity phrases and verbs or preposition as relation phrase.

5.2 Collocation Measure

Once candidate entity mentions are obtained, collocation based filtering is used on the detected phrases that are bi-grams or greater. Some phrases that are detected may not have a good collocation score and such candidates we will removed. In turn, this should help the propagation algorithm type the entity phrase appropriately.

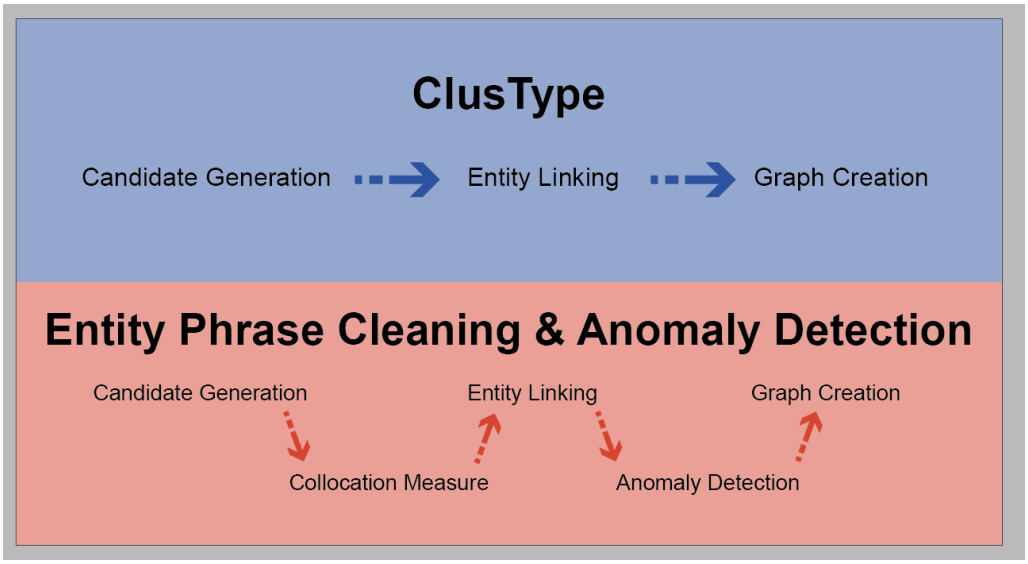

Figure 4: Framework

5.3 Entity Linking

After all the entity phrases are obtained from the candidate generation, they will be mapped to the KB, Freebase. Mapping the candidate entity mentions will typing the ones that have high confidence. This will become the seed entity mention file for the propagation algorithm.

5.4 Anomaly Detection in the Seed Entity Mention

There are two steps to this procedure. It is important to detect anomalies in the seed entity mention file because types that come from the KB could have some erroneous typing since the context of the sentence is not considered. In the anomaly detection, we used Word2Vec, which maps words into low-dimensional space depending on the context of the corpus. Using this vector space features, we used a neural network based Autoencoder algorithm to detect the anomalies.

5.5 Graph Creation

At this stage, we incorporated new improvements to the existing ClusType framework .With the new input data, the ClusType algorithm constructs three different graphs: name-relation, mention-name, and mention correlation subgraph. These graphs are used with the propagation algorithm to propagate the typing in the seed file to the rest of the candidate entity phrases in the text.

6 Experiments

6.1 Data

Our experiments used data from the New York Times, which was collected by crawling news articles in 2013. The dataset contains 118,664 articles (57M tokens and 480k unique words). The New York Times is a new company that covers many aspects of our social life, like US and World politics, finance and business and sports. This is the same dataset that will be used to evaluation ClusType with and without our additions.

In the ClusType algorithm, the data was first filtered to represent the algorithm's purpose better. The data used lemmatization on the tokens so that different forms of words are reduced to lemma form. For example, both "run", "running", "ran" and "runs" are reduced to their lemma word, which is "run".

The seed entity phrases were type with Freebase. The types were converted to represent 4 major categories: person, organization, location, and time/event. The entity phrases were mapped with a name disambiguation tool, and only the entity phrases that had a confidence score over (n >= 0.8) was typed.

Table 1: Anomalies

Organization	christie	hockey	boston
Location	office	mobile	international
Person	boxer	st	rome

Table 2: Results

Algorithm	Precision	Recall	F1-score
ClusType	0.95003743449	0.925373134328	0.937543099202
ClusType+Filtering	0.950419450445	0.91983081336	0.934874987647

For this experiment, we are going to use the subset of documents from the NYT dataset that were annotated. There were 1000 documents annotated for the NYT dataset, which was 25,451 mentions. Each of these mentions are typed to the corresponding types that are in the seed entity file. If the seed entity file has an entity phrase that is in the annotated set, the entity is removed from the seed file, as the test and training data will have the same data, and it will cause error in the evaluation later. Both the ClusType algorithm with and without the additional filtering has the same experimental settings; therefore, we can compare the results with the same parameters and evaluate if the filtering of the entity phrase and seed file produces a more accurate propagation. The experiment uses a maximum phrase length of 5, a minimum support for the phrases in the candidate generation as 30, and the significance threshold as 2. The algorithms are also evaluated at 4000 relation phrase clusters.

6.2 Entity Phrase Collocation Score Results

Entity Phrase Collocation Score - The NYT dataset has a massive raw text and the raw text will have many entity phrases. The collocation measure will remove those phrases before the entity typing and the propagation algorithm is used. The performance of entity detection for ClusType without the collocation score is a precision of 0.469 and a recall of 0.956. From the experiments collocation score based filtering prevents some of the bad entity mentions from typing for example "fla ap jaye marie green" is one entity that is typed as "PERSON" by Clustype framework but with the addition of collocation measure such entities are not typed. This is a valuable reduction in entity phrase detection, as only high quality entity mentions are used for typing.

6.3 Anomaly Detection Results

The anomaly detection is focused on improving the seed entity mentions by removing noisy mentions. Since these are the only typed entity phrases in the propagation algorithm initially, this will affect the accuracy of the results. The Autoencoder was set to [1000,500,20,500,1000] node neutral network and 0.975 quantile of the results from the Autoencoder are within the tolerance to not remove from the seed file. With these parameters, 48 of the seed entity mentions were removed from the seed set and they will not be used in the propagation algorithm. Some of the removed seed entity mentions can be seen in table 1. These seed entity phrase was removed because it could propagate to other entity mentions that are similar to this phrase. The bad seed examples by type can be found in the Anomalies file that is produced by the algorithm.

6.4 Results

After cleaning the inputs for the propagation algorithm, we compared the results of the propagation algorithm without the cleaning and with the cleaning. From Results table the initial results without the cleaning were good, as the algorithm typed entity phrases with 0.95 precision, and 0.93 F1 Score. When the filtering is added to the propagation algorithm, produces a result of 0.93 F1 score. There wasn't as much of an improvement for this text domain because of the high performance on this dataset, see table 2. Through our filtering methods, there are some phrases that are now typed with more meaning and for large corpus fewer entity mentions will be detected but with high quality.

7 Conclusion

The name entity recognition is an important research effort to make text corpus more structure and understandable. Distant supervision has reduced the human effort in entity typing and in turn has allowed the new entity frameworks to work with more text domain and label more entities. The increase in entities has allowed frameworks like ClusType to utilize many seed entity mentions and propagate them through the text corpus. In this paper, we were able to show the importance of entity detection and quality typing for the seed entity mentions. Removing those erroneous mentions improved the propagation to the rest of the text corpus.

References

[1] Xiang Ren, A. El-Kishky, C. Wang, F. Tao, C. R. Voss, H. Ji, J. Han. ClusType: Effective Entity Recognition and Typing by Relation Phrase-Based Clustering ACM SIGKDD Conference on Knowledge Discovery and Data Mining (KDD), 2015

[2] K. Bollacker, C. Evans, P. Paritosh, T. Sturge, and J. Taylor. Freebase: a collaboratively created graph database for structuring human knowledge. In SIGMOD, 2008

[3] GitHub. (2017). dbpedia-spotlight/dbpedia-spotlight. [online] Available at: http://spotlight.dbpedia.org [Accessed 9 Aug. 2017].

[4] Radimrehurek.com. (2017). gensim: topic modelling for humans. [online] Available at: https://radimrehurek.com/gensim/models/word2vec.html [Accessed 9 Aug. 2017].

[5] Xiang Ren*, Wenqi He*, Meng Qu, Lifu Huang, Heng Ji, Jiawei Han. AFET: Automatic Fine-Grained Entity Typing by Hierarchical Partial-Label Embedding Conference on Empirical Methods in Natural Language Processing (EMNLP), 2016.

[6] Manning, Chris; Schütze, Hinrich (1999). Foundations of Statistical Natural Language Processing. Cambridge, MA: MIT Press. pp. 163–166. ISBN 0262133601.

[7] E. Marchi, F. Vesperini, F. Eyben, S. Squartini, B. Schuller, "A Novel Approach for Automatic Acoustic Novelty Detection Using a Denoising Autoencoder with Bidirectional LSTM Neural Networks", Proc. of ICASSP, pp. 1996-2000, Apr. 2015.

[8] Pavel Pecina , Pavel Schlesinger. Combining Association Measures for Collocation Extraction COLING-ACL '06 Proceedings of the COLING/ACL on Main conference poster sessions, 2006.