A Survey Report on ClusType Algorithm and Other Related Methods

Usha Amrutha N, Pulkit Aggarwal, Anahita Hosseini and Deepak M

March 2016

1 Introduction

The Web has, undoubtedly, grown to be one of the largest and crucial data repositories in the world in recent years. A plethora of information is available in the form of natural language. However, deriving meaningful conclusions from large unstructured data is no easy task. One of the main challenges in Natural Language Processing and Information Retrieval domain is to identify the type of named entities. Entity recognition entails Identifying token spans as entity mentions in documents and labelling their types.

For example, consider the following sentence: President Obama arrived in Washington D.C today.
In the above sentence Obama is an entity of type Person, and Washington D.C is entity of type Location. In order to associate a type to entity mention, a Knowledge Base (KB) is required, where all entities are mapped to their respective types. KBs - such as the Wikipedia, DBpedia, FreeBase - provide rich information about semantics and the relationship between entities. However, the information in the Web is so large that, it is humanly impossible to manually curate the type of every entity present in the Internet.

Another downside of having a KB is that not all entities present in the web are linkable to a KB. Nearly, 50 percent of the entities are believed to be un-linkable. Therefore, we need to devise an algorithm such that the un-linked entities are automatically inferred based on the prior observations of entities of similar type .

The basis of this survey report is the paper entitled , "The ClusType: Effective Entity Recognition and Typing by Relation Phrase-Based Clustering" presented in KDD 2015 conference. [1] The Clustype framework aims to overcome the problem of unlinked entities by means of distant supervision learning technique, and also introduces a novel data-driven cluster based entity recognition framework. In this report, we are going to compare various methods that achieve entity recognition.

First, we look into possible ways to approach this problem, and then evaluate ClusType framework against other existing models.

The paper primarily focuses on resolving the above mentioned challenges by exploiting contextual cues provided by relation phrases present in the neighbourhood of the entity, along with a rather novel scheme of clustering relational phrases.
It is interesting to note that the proposed model overcomes the long standing challenges in the text mining domain such as:
1.Domain Restrictions
2.Name Ambiguity
3.Contextual Sparsity.
Subsequent sections are as follows: Section 2 discusses the three challenges mentioned above in detail. In section 3 we discuss the potential ways to approach the unlinked entity mention problem, and review related literature. In Section 4 an we define typical tasks performed by an entity recognition system. Sections 5 and 6 discuss the existing techniques methodology alongside the methods employed in the base paper. In Section 7 we will evaluate Clustype against best compared methods, and the related papers published post KDD 2015.

2 Challenges

2.1 Domain Restriction

Extracting entity mentions is a difficult task mainly because the linguistic features are not consistent among different domains. For instance, articles and magazines have a more formal jargon associated with them as compared to, tweets posted on twitter. Therefore, we cannot have a common knowledge base for entities because they are, by and large, domain-specific. This domain-restriction makes it difficult to mine for appropriate candidate entities and link them efficiently. Most of the designed systems use general domain corpora such as news-reports to extract entity-mentions. However, these days dynamic domains such as tweets and reviews require different approaches. To address this problem [39] has introduced Entity Linking method to detect entity mentions in twitter text and resolving them into entries in Wikipedia, if they exist. This task is done in two steps. The first step is mention candidate generation in which the system takes all n-gram with n less than ten and checks if this n-gram exists in the dictionary or not. The dictionary used here is a mention-entity dictionary with matches a entity-mention to an entity in Wikipedia. To overcome misspelling and irregular forms in twitter text, three approximate string-matching methods: fuzzy match, approximate token search and acronym search has been used. The second step is mention detection and disambiguation. In this part a supervised machine learning model has been used to assign a relevance score to each pair of mention and entity

2.2 Name Ambiguity

When more than one entity mention share same surface name, name ambiguity occurs. For instance, consider the following sentences:

1. Washington is an ally of Kabul.
2. He has his office in Washington
3. Washington beat Ohio by 22-21

In the above examples, we note that the surface name ”Washington” is unique. However, the context in which the word ”Washington” was used differs. That is to say, in sentence 1, Washington is of the type *Government.* Likewise, in examples 2 and 3 the word ”Washington”, refers to type *place* and *Sports team.* In order to disambiguate similar entities, we need to exploit the contextual information present in the text corpus to infer the type of the entity mention

2.3 Contextual Sparsity

Certain phrases, though synonymous to commonly used phrases,are often infrequent. This is a difficult challenge, especially, when the model extracts entity types based on the contextual relation phrases as well. observe the following sentences, and their frequencies.

1.Earthquake caused widespread devastation in Southern California - Frequency (250) .
2.Earthquake ravaged Southern California (19)

We observe that the sentence ”widespread devastation” is more commonly used than ”ravaged”, though both phrases link same entity-type ” Disaster” and ”Place” together. In order to be able to derive entity type effectively, it is important that even the infrequent phrases are used to deduce the entity mention - type relations. [40] has proposed a new fully language-independent approach in Named Entity Recognition. To make this possible, they have used an unsupervised machine learning method to learn word similarities and cluster them based on semantic space. Features fed into this model are semantic features that are automatically created and enriched with stemming approaches. The final named entity recognition task has been done by means of a conditional random fields to create annotated text data

3 Approach

The problem of entity linking can be broadly approached in two ways : 1. Weak Supervised Learning. 2. Distant Supervised Learning.

3.1 Weak Supervised Learning

Weak supervised learning is also known as semi-supervised learning (SSL), where the patterns are learned from large unlabeled text as well as small amount of labeled data. Methods such as the Bootstrapping, have employed weak supervised learning techniques for entity extraction. In this method we set initial seed values of entities and learn patterns from the unlabeled text and a set of rules defined by lexico-syntatic surface-word patterns[2] or dependency tree patterns[3]. [2] has tried to improve this method by introducing a pattern ranking algorithm. Basically, the algorithm works by generating new patterns and subsequently finding new entity relations from the text. Candidate patterns are generated using the saved entity relations in the dictionary as seeds. Patterns thus generated are assigned pattern scores. The top ones are added to the learned patterns of the class, and are applied to the text to further add more entity relations to the dictionary. This method is performed iteratively until no entities can be learned further.

In [4], authors devised another bootstrapping approach to induce semantic class taggers from domain text. The semantic class tagging process of un-annotated entities takes place in two steps. Firstly,all seed words are blind-folded such that a classifier can only see the context around them. A set of strictly contextual classifiers created every semantic category, decides whether or not a noun phrase belongs to a particular semantic category. When applied to the corpus, the contextual classifiers automatically labels unlabeled data to generate new instances based on contextual cues. These labeled instances are added to the training data-set. This method improves diversity of the training data to a large extent as it contains both seed-generated and context- generated instances. In the phase 2 of learning process, Cross-category bootstrapping is performed, where a set of binary classifiers are trained for each semantic category. If a set of new instances is positively labelled by category, say, Ck, we add the new instances to the positive training set for classifier Ck, and at the same time add them to the negative training set for all of the other classifiers. Therefore, multiple classifiers for all semantic categories are trained simultaneously. Like in the previous bootstrapping strategy, it is performed iteratively until no entities can be learned.

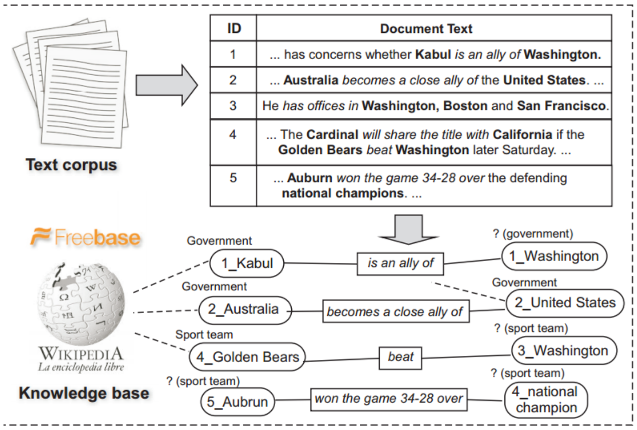
One of the setbacks of this procedure is that the cross-category bootstrapping will not work very well if a particular semantic category is more dominant than the rest. In this case, the positive and negative training instances of every other class may be imbalanced if one class appears more frequently than the other. Also, the semantic classes related to noun phrases must be established prior to the bootstrapping method. [5] Method have eliminated the necessity of relying on NP, by focusing on the word - level name disambiguation

The disadvantages of the weak supervised learning methods are that it places undue importance on the seed information. If the seeds of a particular type aren't sufficiently frequent, the model suffers from severe degradation. Moreover, the methods assume that the seed words are unambiguous. In reality as discussed above, name disambiguation remains to be one of the critical challenges in NLP. These unrealistic assumptions make weak supervised learning techniques unfit for large corpus entity-linking tasks. Other weak-supervised learning technique is Label Propagation (LP), which is discussed in detail in Section 6.2.

3.2 Distant Based Supervision Learning

Distant supervision learning is a recent trend, which aims to minimize the human intervention as much as possible. This approach makes use of the information about entity types present in the KBs, and infer the type of unlinked entities from the similar type present in KB. The work flow of a typical distant based supervision learning is as follows:
1. Detect entities from the text corpus
2. Map the known entities to the type present in KB
3. Use the confidently mapped pairs entities, type to predict the type of unlabeled entity.

Fig1 demonstrates distant-based supervision learning related tasks In [6] the distant supervision learning is used to disambiguate different types of same entity. The main idea is to bridge the gap between Freebase and Knowledge Base, and use information present in both to make intelligent type predictions on unlinked mentions. The novelty of this work lies in the way which the labels are transformed into features before feeding it to the classifier, in order to prevent creation of individual classifier for a huge set of ambiguous entities. Bootstrapping methods that were discussed in previous section usually suffer from semantic drift and as a result low precision. Moreover, relations achieved by employing completely unsupervised learning cannot usually be applied when using a specific knowledge base with a particular context. The assumption of this method is that, when a relation is found between two entities, those entities in any other sentence may also show that relation. The basic idea behind this paradigm is definition of feature vector for each relation between two entities. Features extracted are mostly lexical features such as the sequence of words



ID	Document Text
1	... has concerns whether **Kabul** *is an ally of* **Washington.**
2	... **Australia** *becomes a close ally of* the **United States**. ...
3	He *has offices in* **Washington, Boston** and **San Francisco**.
4	... The **Cardinal** *will share the title with* **California** if the **Golden Bears** *beat* **Washington** later Saturday. ...
5	... **Auburn** *won the game 34-28 over* the defending **national champions**. ...

Figure 1: Typical flow of distant supervision

between the two entities or A window of k words to the left of Entity 1 and their part-of-speech tags. Along with this, a number of syntactic features are also extracted. In the training phase, features for identical relations are extracted and combined as feature vector. Then in testing phase, again feature extraction from all identical relation phrases is done and participating entities are predicted relation name based on features.

In [7] authors proposed a model for discovering and ontologically typing entities not present in the KB, using a fine-grained type system and by exploiting relational paraphrases with type signatures using probabilistic models. This is in line with the ClusType framework as both of them share the same paradigm. However, the ClusType addresses other challenges of contextual sparsity and name disambiguation along with predicting the type of unlinkable mentions.

ClusType framework, on the other hand, employs distant supervision method by solving a joint optimisation problem conditioned on mention names, surface names, type signature of relational phrases - details of which we shall discuss in the subsequent sections.

4 A typical entity- recognition framework

A general entity- recognition / typing framework consists of the the following modules:
1. **Candidate Entity Generation**: The idea behind candidate generation is to filter entity mentions which are either of low quality or insignificant.This is an important step in any entity-linking system in order to get rid of irrelevant information present in the text corpus. A variety of techniques used by some

state-of-the-art entity linking systems involve name dictionary based techniques, surface form expansion from the local document, and methods based on search engine. In CLUSType framework [1], a novel phrase-mining algorithm has been proposed to generate entity candidates which have minimalistic dependency on the linguistic/ domain assumption.

2.**Candidate Entity Learning** : This module of the entity recognition system typically focuses on mapping a given entity to its type. The ClusType primarily focuses on a graph-based approach for achieving this task. Since only some entities are mapped with prior information present in KB, ranking in ClusType is based on semi-supervised learning, and it constructs various heterogeneous graphs to accomplish this task. In subsequent sections we elaborate more on how other systems have performed this task using supervised, unsupervised and semi-supervised learning methods.

3. **Unlinked Mention Prediction**: This is the principal focus of the survey report. The weak and distant supervised learning methods are the top approaches to circumvent the problem of unlinked mention prediction. As mentioned ClusType uses the distant based supervised learning to predict unlabeled mentions. In section 3 we have already discussed about the possible approaches to deal with this problem.

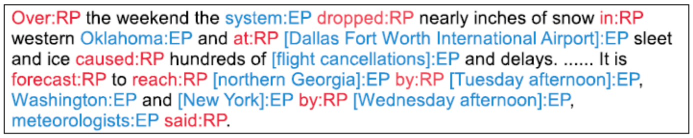
5 Candidate Entity Generation

Firstly, we would like to discuss the method proposed in the ClusType algorithm.This section discusses about the various candidate generation techniques that have been proposed so far in entity linking systems.

5.1 Candidate Generation in ClusType

. ClusType framework uses a novel phrase mining algorithm to generate candidate entity mentions. The idea is to generate candidates independent of the domain and linguistic features. This turns out to be crucial in resolving domain-specific challenges. For instance, this algorithm generates candidates effectively, irrespective of whether it is belongs Yelp reviews corpora or Tweets.

The phrase mining algorithm mines for a fixed length of frequent contiguous patterns. The counts of such occurrences are aggregated, and a greedy agglomerative merging is then performed to form longer phrases. while this happens, certain syntactic correctness constraints are imposed. Finally, a global significant score is defined to filter candidates of low quality below a certain threshold. Along with this generic POS patterns are used to remove relational phrases



Over:RP the weekend the system:EP dropped:RP nearly inches of snow in:RP western Oklahoma:EP and at:RP [Dallas Fort Worth International Airport]:EP sleet and ice caused:RP hundreds of [flight cancellations]:EP and delays. ...... It is forecast:RP to reach:RP [northern Georgia]:EP by:RP [Tuesday afternoon]:EP, Washington:EP and [New York]:EP by:RP [Wednesday afternoon]:EP, meteorologists:EP said:RP.

EP: entity mention candidate; RP: relation phrase

Figure 2: Typical flow of distant supervision

which do not comply with the syntactic constraints. The text corpus is then partitioned into segments which meet both POS patterns and Global significance scores thresholds. The phrase merging step selects the most significant merging, by comparing the frequency of a potential merging of two consecutive phrases to the expected frequency assuming independence. Figure 2 shows generation of candidate entity mentions in relation phrases.

5.2 Candidate Generation in other entity-linking systems

In [8 9 10 11] researchers use name dictionary based techniques. Though only a few papers have been cited here, we found that a lot of researchers have preferred to employ this technique in entity candidate generation. The main idea is to ensure all the candidates that have been referred to in more than one form are represented in a standard manner. For instance, "Google" can be referred as "Google Inc" or "Google Corporation". Therefore, a name dictionary is maintained in such a way that a key,value table provides entity name, mapping entity. Therefore, when an entity name is encountered it is checked against the name dictionary, and the corresponding mapping entity is extracted. Some methods consider only partial matching - entities that share common words in the entity mention, first few letters of a word etc. Other approaches include heuristic pattern matching [13 14], where the entities are expanded based on textual context around the entity mention. Some researchers have leveraged supervised learning methods, which have been proved very useful for acronym expansion. For instance, UCLA - University of California Los Angeles, forms an acronym-expansion pair. Their approach was to apply SVM classifier to each candidate acronym-expansion pair. The highest output confidence score decides the most suitable expansion for a given acronym.

Another emerging form of entity candidate generation is querying the web based search engine about entity mentions to derive the candidate entities. The ClusType algorithm uses this approach to derive candidate entities by querying the Wikipedia search engine. [15 16] have used Google APIs to query entity mentions, and the top 20 results from wikipedia are used to derive the candidate entity mentions.

6 Candidate Entity learning

Like in the previous section, we will give a brief overview of graph-based approach along with relation-phrase clustering mechanism proposed by the ClusType framework, followed by a survey on learning methods used in NER systems.

6.1 Graph based learning and Relation Phrase clustering in ClusType framework

ClusType takes a heterogeneous graph-based approach to represent the candidate entity mentions present in the text corpus in a single unified framework. The synonymous relation phrases are combined together by a soft clustering mechanism to avoid problems that arise due to contextual sparsity. The ClusType makes use of surface names, which is unique to multiple entities which have same entity mention names, but different entity types associated with it. For instance, Washington is a unique surface name for all entity mentions, where Washington could be used in different context , say, with place, government and sports team as its entity mention types. Three sub-graphs are constructed based on the following objects
1. Candidate Entity Mentions
2. Entity Surface Names
3. Relation Phrases

6.1.1 Entity Mention- Surface Name sub-graph

This graph is constructed to establish a relation between every entity and its unique surface name. The purpose of this is to obtain the distribution of surface names in the given text corpus. Another striking aspect of the ClusType framework is that, it exploits the presence of the relational phrases to derive contextual cues, and thereby identify type of each entity mention. Each relation phrase is represented by a left type argument and right type argument, which contains the type of candidate entities present to the left and right positions of a relation phrase. An example of type signature is shown below.
Type Signature of RP: [Type left argument.], [Type Right argument.]
E.g.:Kabul is an ally of Washington
“ is an ally” : [Kabul: Government], [Washington: Government]

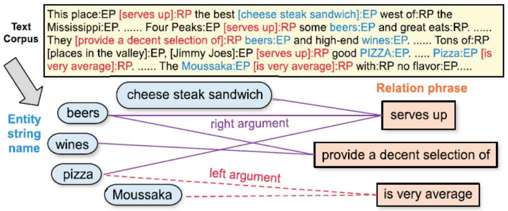

Text
Corpus
This place:EP [serves up]:RP the best [cheese steak sandwich]:EP west of:RP the
Mississippi:EP. ...... Four Peaks:EP [serves up]:RP some beers:EP and great eats:RP. ......
They [provide a decent selection of]:RP beers:EP and high-end wines:EP. ...... Tons of:RP
[places in the valley]:EP, [Jimmy Joes]:EP [serves up]:RP good PIZZA:EP. ..... Pizza:EP [is
very average]:RP. ...... The Moussaka:EP [is very average]:RP with:RP no flavor:EP.....
Entity
string
name
cheese steak sandwich
beers
wines
pizza
Moussaka
Relation phrase
serves up
provide a decent selection of
is very average
right argument
left argument

Figure 3: Entity name-relation phrase subgraph

6.1.2 Name - Relation sub-graph

In order to establish a relation between entities and relation phrases, a name relation - subgraph is constructed. This graph is based on the Entity-Relation co-occurrences hypothesis, which states that [1]:

Hypothesis 1 (Entity - Relation Co-occurrences)if surface name c often appears as the left (right) argument of relation phrase p, then c's type indicator tends to be similar type indicator in p's type signature.

for example, if we know that "pizza" of type food frequently co-occurs as right argument with relations phrase "serves up", then any other entity mention that appears as right argument to the relation- phrase is likely to be of type food. By this argument "cheese-steak sandwich" is identified of as entity of type food. The Fig 3 provides the graphical representation of hypothesis stated above.

6.1.3 Mention-Mention relation subgraph

This sub-graph primarily aims to disambiguate between the entity types which share common entity surface name. We use an affinity sub graph to represent the mention-mention link based on k-nearest neighbor (KNN) graph construction.

Hypothesis 2 (Mention correlation). If there exists a strong correlation (i.e., within sentence, common neighbor mentions) between two candidate mentions that share the same name, then their type indicators tend to be similar

Fig 4 shows a typical mention-mention links. In this example, the context in which the word " White House" appears is observed over k-nearest neighbours. It is likely that the entities which share strong co-relation, defined by similarity score in [1], have same entity type. Therefore, Fig 4 demonstrates two distinct sub graphs of the same entity "White House" which appears in two different contexts (with respect to type place and government).

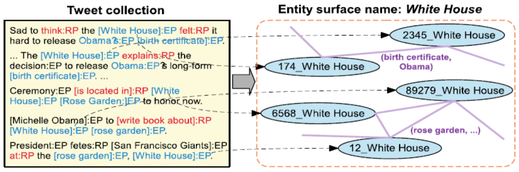

Tweet collection
Sad to think:RP the [White House]:EP felt:RP it hard to release Obama?s:EP [birth certificate]:EP.
... The [White House]:EP explains:RP the decision:EP to release Obama:EP?s long-form [birth certificate]:EP. ...
Ceremony:EP [is located in]:RP [White House]:EP [Rose Garden:]EP to honor now.
[Michelle Obama]:EP to [write book about]:RP [White House]:EP [rose garden]:EP.
President:EP fetes:RP [San Francisco Giants]:EP at:RP the [rose garden]:EP, [White House]:EP.
Entity surface name: White House
2345_White House
174_White House
(birth certificate, Obama)
89279_White House
6568_White House
(rose garden, ...)
12_White House

Figure 4: Mention-mention link subgraphs

6.1.4 Relation-Phrase Clustering

In order to avoid contextual sparsity the Clustype jointly clusters synonymous phrases using soft-clustering mechanism. In the experimental data used in this work, more than 37 percent of the data contain sparse relational phrases. The Clustype framework infers type signature of infrequent (sparse) relational phrases, using type signature of frequent relational phrases. This is based on the hypotheses stated as follows:

Hypothesis 3: (Type Signature Consistency):if two relation phrases have same cluster memberships, their type signatures tend to be similar

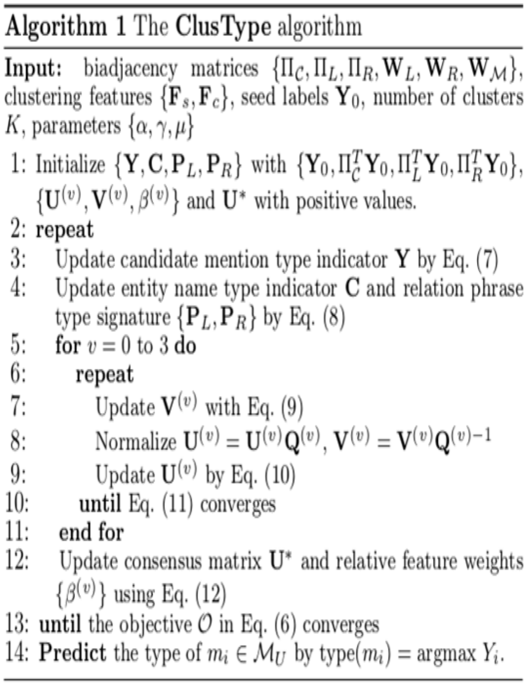
Hypothesis 4: (Relation phrase similarity) : Two relation phrases tend to have similar cluster memberships, if they have similar (1) strings; (2) context words; and (3) left and right argument type indicators

Their approach uses the derived features (i.e., Fs, Fc, PL, PR) for multi-view clustering of relation phrases based on joint non-negative matrix factorization. Where, Fs and Fc represent extracted features of string and context. PL , PR represent the left and right argument vector of type signature of a relation phrase.

In Clustype Learning two optimization tasks occur simultaneously: 1.graph-based semi-supervised learning for type propagation over type indicators, and type signatures of relation phrases. 2. Multi-view relation phase clustering

This is the first work in NER domain to combine phrase clustering and graph-based learning as the type propagation occurs. The Fig 5 gives an brief insight into the working of ClusType. Basically, it follows an iterative approach to solve the join optimisation problem until the objective function (input parameters to the Clustype algorithm) converges. The input parameters such as the type indicator matrices are updated after each iteration. While this occurs, within each iteration, the relation-phrase clustering is performed on four different views discussed earlier. Then they define a consensus matrix that ensures that synonymous phrases clustered within a group are consistent from all four views. This optimisation problem is solved until the objective function converges.

The most important step of the ClusType is highlighted in line 14 of the algorithm shown in the figure. After the clustering and type propagation is complete,



```
**Algorithm 1** The **ClusType** algorithm
**Input:** biadjacency matrices $\{\Pi_{\mathcal{C}}, \Pi_L, \Pi_R, \mathbf{W}_L, \mathbf{W}_R, \mathbf{W}_{\mathcal{M}}\}$, clustering features $\{\mathbf{F}_s, \mathbf{F}_c\}$, seed labels $\mathbf{Y}_0$, number of clusters $K$, parameters $\{\alpha, \gamma, \mu\}$
Initialize $\{\mathbf{Y}, \mathbf{C}, \mathbf{P}_L, \mathbf{P}_R\}$ with $\{\mathbf{Y}_0, \Pi_{\mathcal{C}}^T\mathbf{Y}_0, \Pi_L^T\mathbf{Y}_0, \Pi_R^T\mathbf{Y}_0\}$,
    $\{\mathbf{U}^{(v)}, \mathbf{V}^{(v)}, \beta^{(v)}\}$ and $\mathbf{U}^*$ with positive values.
**repeat**
    Update candidate mention type indicator $\mathbf{Y}$ by Eq. (7)
    Update entity name type indicator $\mathbf{C}$ and relation phrase
        type signature $\{\mathbf{P}_L, \mathbf{P}_R\}$ by Eq. (8)
    **for** $v = 0$ to 3 **do**
        **repeat**
            Update $\mathbf{V}^{(v)}$ with Eq. (9)
            Normalize $\mathbf{U}^{(v)} = \mathbf{U}^{(v)}\mathbf{Q}^{(v)}$, $\mathbf{V}^{(v)} = \mathbf{V}^{(v)}\mathbf{Q}^{(v)-1}$
            Update $\mathbf{U}^{(v)}$ by Eq. (10)
        **until** Eq. (11) converges
    **end for**
    Update consensus matrix $\mathbf{U}^*$ and relative feature weights
        $\{\beta^{(v)}\}$ using Eq. (12)
**until** the objective $\mathcal{O}$ in Eq. (6) converges
**Predict** the type of $m_i \in \mathcal{M}_U$ by type$(m_i)$ = argmax $Y_i$.
```

Figure 5: ClusType algorithm paradigm

if a given entity in unlinked in the KB, the argument with maximum value in type indicator matrix Y is assigned as the predicted type for the given candidate entity mention. This way the algorithm infers the types of entities that are unlinked to KB.

6.2 Other learning methods used in entity recognition

Clustype primarily focuses on graph-based semi-supervised learning to achieve name disambiguation as well as learning the entity type . In the traditional learning methods, both supervised and unsupervised learning methods have been proposed. classical supervised techniques [17 18 19], typically glean through the large annotated text corpus, and learn all entities present in the corpus,and create a set rules for disambiguation based on the discriminative features. Typical supervision based learning models such as Hidden Markov Model (HMM) [17], Maximum Entropy Models (ME) [18], Support Vector Machines (SVM) and Conditional Random Fields(CRF) [19] have been used in the task of named-entity recognition. HMMs and MEs are probabilistic models. While HMMs is a generative model which generates all words and labels based on distribution parameters, MEs are discriminative models that maximise the entropy so as

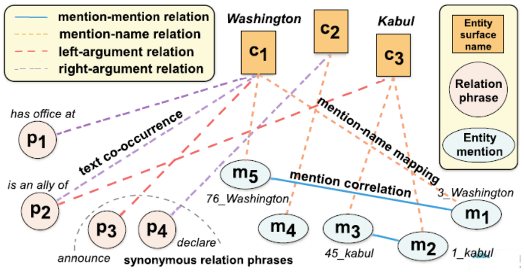


Figure 6: ClusType entity recognition framework

to generalise as much as possible for the training data. CRF , on the other hand, is a probabilistic graphical model that is used widely in case of structured predictions. Whereas, classifier like SVM predict whether or not a candidate entity belongs to a particular type, the CRF models work on sequence of inputs to predict a sequence of outputs. However, the above methods may not be effective especially if the text corpus is improperly annotated, which is often the case with the web in recent times. Also, unsupervised learning methods such as clustering techniques [20 21] group entities of similar type together. From the problem definition it is evident that a entity mention can belong to more than one type. In such a scenario, the clustering mechanism may not work effectively. The semi-supervised learning (SSL) methods , such as bootstrapping and labeling methods which have been discussed at length in Section 3 are some of the other learning methods, which work on small set of labelled and large amount of unlabeled data.

Besides that, other SSL methods such as co-training [22] , transductive support vector machines (SVMs)[23] , and graph-based SSL [24], have been some of the successful approaches in SSL domain related to NER problems. Though SSL methods have proven to be fairly accurate, it is computationally very expensive. In practice, there has to be an ideal trade-off between accuracy and computational complexities. Some researchers have dedicated their efforts to address this conflict, however, they led to suboptimal results [25]. Also, all of the SSL methods mentioned previously do not scale well with large text corpus. Scalability is yet another important aspect to be considered when dealing with SSL based approaches for large amount of data. In this regard, graph-based SSL methods have proven to be easily scalable [26]. In graph-based approach vertexes are candidate entity mentions, which could be both labeled and unlabeled. The links are established between those vertexes are likely to have the same label. Like in any other graph model, edge-weights represent how strongly the labels are connected.

In [27], authors proposed one of the first graph-based SSL algorithms based on Label Propagation (LP). The idea is to minimise the objective function, by enforcing certain constraints that ensure supervised labels are not changed during inference, or as new labels are being propagated. Another important aspect in LP method is that it penalises those nodes which assign high edge weights to the link which connect nodes of different labels. Another notable work in graph-based SSL domain is [28], where the authors proposed an Adsorption algorithm. They leveraged an iterative apporach, where the labels estimated on node in (k+1) th iteration are updated based on the label estimate of kth iteration. Every node is assigned three probabilities that sum to one. The probability model is derived from the random-walk interpretation of the Adsorption algorithm. The crux of Adsorption is to control the amount of information that flows through nodes. Unlike LP technique, it places a loose constraint on the seed labeled nodes during inference, which is highly desirable in presence of noisy labels. However, Adsorption does not have a well-defined objective function. Therefore another graph-based SSL method Modified Adsorption (MAD) algorithm was proposed [29] which is very similar to the Adsorption algorithm, expect the fact that it is expressed as unconstrained optimisation problem solved using an iterative approach. According to the experimental survey conducted by [29] on FreeBase-1 and TextRunner , MAD has proven to be the best among the three graph-based SSL methods mentioned above.

Another graph-based SSL model proposed recently in NER domain for unstructured data includes [30], where the graphs are used in conjunction with CRF learning models. According to us, the results are sub-optimal compared to the state-of-art entity recognition systems primarily because CRF-based approach may be more accurate for structured prediction, and does not work well with unstructured models. A recent work on [31] structured tagging models is yet another graph-based SSL method, but primarily used for structured corpus.

For ill-structured domains such as twitter, text is subject to misspelling and other forms of noise. However as mentioned earlier, many supervised methods used just give good results for well-structured and large data-sets. [41] has proposed a new approach for addressing the problem of name-entity recognition on twitter. In their work, they use a semi-supervised approach along with Conditional Random Fields(CRF) and follow three main steps to achieve desired performance. Firstly, they use clustering methods, to put unlabeled entities in different groups. Then CRF model is employed to train a classifier based on labeled tweets and finally, by using co-occurrence coefficients, performance is improved. Each time new labeled entities are added to the training set. In this work they argue that CRF gives lower recall than desired that is why they use a co-occurrence coefficient for finding named-entity candidates. The coefficient is calculated based on parameters such as distance from feature word and number of occurrences in sentences. They show that with this approach they can achieve satisfying results even when data set size is small.
Having looked into some of the popular learning methods - supervised, unsupervised and semi-supervised learning models - employed in NLP domain, in the next section we would like critically evaluate the ClusType framework.

7 Evaluation of Clustype with other compared methods

ClusType is efficiently solved by alternate minimization based on block coordinate descent algorithm. The algorithm complexity is linear to the number of entity mentions, relation phrases, cluster, clustering features and target types. The variants of the Clustype only model a part of the hypotheses proposed in the paper. Some of the comparable approaches are Stanford NER [32], which is a CRF Classifier trained on corpora with large number of entities.. FIGER [33] trains sequence labeling models using Wikipedia corpora. It makes use of linguistic features to train sequence labeling models but suffers from low recall in the presence of noisy data like Tweets. Therefore, the feature generation method in FIGER does not work well. In comparison to FIGER, ClusType obtained 46.08 percent improvement on F1 score and 168 percent improvement in recall compared to best baseline of FIGER on Tweet datasets. Pattern-based bootstrapping discussed in Section 2 and SemTagger [4] - use seed mention set for self-training. However, Clustype performs better than bootstrapping methods as it works well even if the seed mentions are sparse in text corpora. This is in contrast with bootstrapping methods which assume that seed mentions are sufficiently large. NNPLB [34] uses ReVerb Open IE system [35] - which contains uninformative and incoherent text- to construct graphs and performs entity name-level label propagation based two simple lexical and syntactic constraints. The F1 and precision score obtained by Clustype are far superior to NNPLB. Also, NNPLB does not consider the name-ambiguity problem, and thus suffers when multiple entities share same surface name. Another method APOLLO [36] builds heterogeneous graphs on entity mentions, KB entities, and then performs label propagation. Yet again, this models only a part of the problem statement under consideration, and does not take into account the problem of contextual sparsity.

However, Clustype , like many other NER systems, ignores the label noise present in the automatically trained corpora. A more recent work [37], is centered on building a Partial Embedded labelling (PLE) framework which takes into account the noisy labels. Like ClustType, PLE constructs a heterogeneous graph to represent entity mentions, text features and entity types, and their relationships within a framework. Then, they formulate joint optimisation problem, and in order to comply with limitations posed due to high-dimensionality, they jointly embed the graph in a low-dimensional space. The objects in this space that are semantically similar to one another, share same representation. In order to overcome the problem of noisy data and false candidates, PLE framework proposes margin-based rank loss to model entity candidate entity mention-type.This model ensures that only the best candidate type is embedded close to the mention in lower dimensional space, thereby eliminating false associations of noisy labels. Another, recent work [38] in line with Clustype uses genetic algorithm for unsupervised medical term extraction from clinical letters. Without domain specific knowledge, the paper uses key phrase extraction based on co-occurrence graphs, prefix span sequential pattern mining, and linguistic features extracted from C-value. Since, the three methods represent different linguistic features, a genetic algorithm was used to learn the best parameters.

8 Conclusion

In this report, we defined the problem statement with respect the Clustype algorithm. Different methods to approach the problem of predicting unlinked entity mention were discussed, namely weak-supervised and distant supervision learning. Later, for every module of the framework proposed we compared it to the techniques employed by the other state-of-the-art methods. Various candidate entity generation techniques were also discussed at depth along with existing learning methods. We finally, evaluated Clustype to other methods in terms of F1, precision and recall scores. In addition, we also conducted survey on papers that had similar problem statement that were published recently/submitted to KDD 2016.

9 References

[1] X. Ren, A. El-Kishky, C. Wang, F. Tao, C. R. Voss, and J. Han, "Clustype: Effective entity recognition and typing by relation phrase-based clustering," in KDD, 2015, pp. 995–1004.

[2] S. Gupta and C. D. Manning. Improved pattern learning for bootstrapped entity extraction. In CONLL, 2014

[3] Winston Lin, Roman Yangarber, and Ralph Grishman. 2003. Bootstrapped learning of semantic classes from positive and negative examples. In Proceedings of the ICML 2003 Workshop on The Continuum from Labeled to Unlabeled Data in Machine Learning and Data Mining.

[4] R. Huang and E. Riloff. Inducing domain-specific semantic class taggers from (almost) nothing. In ACL, 2010.

[5] T. Lin, O. Etzioni, et al. No noun phrase left behind: detecting and typing unlinkable entities. In EMNLP, 2012.

[6] Fan, M.; Zhou, Q.; and Zheng, T. F. 2015. Distant supervision for entity linking

[7] N. Nakashole, T. Tylenda, and G. Weikum. Fine-grained semantic typing of emerging entities. In ACL, 2013.

[8] X. Han and L. Sun, “A generative entity-mention model for linking entities with knowledge base,” in ACL, 2011, pp. 945– 954.

[9] S. Gottipati and J. Jiang, “Linking entities to a knowledge base with query expansion,” in EMNLP, 2011, pp. 804–813.

[10] K. Chakrabarti, S. Chaudhuri, T. Cheng, and D. Xin, “A framework for robust discovery of entity synonyms,” in SIGKDD, 2012, pp. 1384–1392

[11] B. Taneva, T. Cheng, K. Chakrabarti, and Y. He, “Mining acronym expansions and their meanings using query click log,” in WWW, 2013, pp. 1261–1272

[12] W. Zhang, Y. C. Sim, J. Su, and C. L. Tan, “Nus-i2r: Learning a combined system for entity linking,” in TAC 2010 Workshop, 2010.

[13] Z. Chen, S. Tamang, A. Lee, X. Li, W.-P. Lin, M. Snover, J. Artiles, M. Passantino, and H. Ji, “Cuny-blender tac-kbp2010 entity linking and slot filling system description,” in TAC 2010 Workshop, 2010.

[14] J. Lehmann, S. Monahan, L. Nezda, A. Jung, and Y. Shi, “Lcc approaches to knowledge base population at tac 2010,” in TAC 2010 Workshop, 2010

[15] W. Zhang, J. Su, C. L. Tan, and W. T. Wang, “Entity linking leveraging automatically generated annotation,” in COLING, 2010.

[16] M. Dredze, P. McNamee, D. Rao, A. Gerber, and T. Finin, “Entity disambiguation for knowledge base population,” in COLING, 2010, pp. 277–285

[17] Daniel M. Bikel, Richard Schwartz, and Ralph M. Weischedel. An algorithm that learns whats in a name. Mach. Learn., 34(1-3):211–231, feb 1999

[18] Andrew Eliot Borthwick. A maximum entropy approach to named entity recognition. PhD thesis, New York, NY, USA, 1999.

[19] Andrew McCallum and Wei Li. Early results for named entity recognition with conditional random fields, feature induction and web-enhanced lexicons. In Proceedings of the seventh conference on Natural language learning at HLT-NAACL 2003 - Volume 4, CONLL '03, pages 188–191, Stroudsburg, PA, USA, 2003.

[20] Alfonseca, Enrique; Manandhar, S. 2004. An Unsupervised Method for General Named Entity Recognition and Automated Concept Discovery. In Proc. International Conference on General WordNet.

[21] Roman Yangarber, Winston Lin, and Ralph Grishman. 2003. Unsupervised learning of generalized names. In Proceedings of the 19th International Conference on Computational Linguistics.

[22] A. Blum and T. Mitchell. 1998. Combining labeled and unlabeled data with co-training. In COLT: Proceedings of the Workshop on Computational Learning Theory.

[23] R. Collobert, F. Sinz, J. Weston, L. Bottou, and T. Joachims. 2009. Large scale transductive svms. Journal of Machine Learning Research.

[24] J. Blitzer and J. Zhu. 2008. ACL 2008 tutorial on Semi-Supervised learning.

[25] O. Chapelle, B. Scholkopf, and A. Zien. 2007. SemiSupervised Learning. MIT Press.

[26] A. Subramanya and J. A. Bilmes. 2010. Entropic graph regularization in non-parametric semi-supervised classification. In Neural Information Processing Society (NIPS), Vancouver, Canada, December.

[27] X. Zhu, Z. Ghahramani, and J. Lafferty. 2003. Semisupervised learning using gaussian fields and harmonic functions. ICML-03, 20th International Conference on Machine Learning.

[28].S. Baluja, R. Seth, D. Sivakumar, Y. Jing, J. Yagnik, S. Kumar, D. Ravichandran, and M. Aly. 2010. Video suggestion and discovery for youtube: taking random walks through the view graph.

[29] P. P. Talukdar and F. Pereira. Experiments in graph-based semi-supervised learning methods for class-instance acquisition. In ACL, pages 1473–1481, 2012.

[30] Aaron and Li-Feng Han and Xiaodong Zeng and Derek F. Wong and Lidia S. Chao, Chinese Named Entity Recognition with Graph-based Semi-supervised Learning Model, 2015.

[31] A. Subramanya, S. Petrov, and F. Pereira. 2014 Efficient Graph-based Semi-Supervised Learning of Structured Tagging Models.

[32] J. R. Finkel, T. Grenager, and C. Manning. Incorporating non-local information into information extraction systems by gibbs sampling. In ACL, 2005.

[33] X. Ling and D. S. Weld. Fine-grained entity recognition. In AAAI, 2012.

[34] T. Lin, O. Etzioni, et al. No noun phrase left behind: detecting and typing unlinkable entities. In EMNLP, 2012

[35] A. Fader, S. Soderland, and O. Etzioni. Identifying relations for open information extraction. In EMNLP, 2011.

.

[36] W. Shen, J. Wang, P. Luo, and M. Wang. A graph-based approach for ontology population with named entities. In CIKM, 2012.

[37] Xiang Ren, Wenqi He, Meng Qu, Clare R. Voss, Heng Ji, Jiawei Han, Label Noise Reduction in Entity Typing by Heterogeneous Partial-Label Embedding', submitted to KDD 2016.

[38] Wei Liu, Bo Chuen Chung, Rui Wang, Jonathon Ng, Nigel Morlet, A genetic algorithm enabled ensemble for unsupervised medical term extraction from clinical letters. December 2015. Conference on Health and Information Systems. Decenber

[39] Yamada123, Ikuya, Hideaki Takeda, and Yoshiyasu Takefuji. "Enhancing Named Entity Recognition in Twitter Messages Using Entity Linking." ACL-IJCNLP 2015 (2015): 136.

[40] Konkol, Michal, Tomáš Brychcín, and Miloslav Konopík. "Latent semantics in named entity recognition." Expert Systems with Applications 42.7 (2015): 3470-3479.

[41] Tran, Van Cuong, Dosam Hwang, and Jason J. Jung. "Semi-supervised approach based on co-occurrence coefficient for named entity recognition on Twitter." Information and Computer Science (NICS),. IEEE, 2015.