Diversified Named Entity Recognition for Web Search Query

Ruiqiang Zhang **Youssef Billawa** **Yi Chang**

Abstract

This paper propose using gradient boosted decision tree (GBDT) method in natural language processiong to attack named entity recognition (NER) on Web search query task,. This approach performs significant different results comparing to the well-known CRF method. While the CRF produces the best top 1 NER, we found the CRF was inferior to the GBDT in dealing with ambiguous queries with multiple intents. Due to query intent ambigurity, the CRF-based NER generates less diversified results and has lower precision for queries with multiple intents. The proposed GBDT method can achieve better diversity results because of global features, subjective judgement as training targets and multiple sources to generate named entities. Our experiments prove the GBDT signifiantly improved NER accuracy.

1 Introduction

Query is ambiguous. Web search users usually send short and ambiguous queries because they want to get fast search results. Search engines have to adapt to such incomplete queries to guess search intents. "Apple" can mean different things by user. It could be a fruit (Food), or apple product (Brand) or apple corporate (Business). Discovering query intent is an important factor for improving Web search user experience. All the ambiguous query intents should be detected by search engines. A good search engine can give diversified search results. She can understand and translate queries into multiple categories. Top search engines such as Google, Bing and Yahoo can present users with results from different categories. Some typical categories include Product, Shopping, Travel, Local, etc. These category search results are positioned on the eye-striking places on the on search result page. They are called Direct Display (DD) by some search engines. DD Search retrieves content from storages of a specific category. Therefore, given their important layout, understanding query intent is critical in these tasks.

NER is one of the big techniques in query intent understanding. Most published work on NER is related to text-based. But query NER is quite different from text. The task difficulty of query NER is higher than that of text NER. This can be seen from calculating language model's perplexity. Perplexity of trigram language model in our work on Web search queries is 385, while the value is 141 for normal English text (evaluated by WSJ treebank (Mikolov and Zweig, 2012)). Higher perplexity is because query is ambiguous, non-grammatical, and less contextual information available. These are reasons why query NER have lower accuracy than text NER. Diversity is a new problem for query NER. There are many cases that queries have multiple intents. All entity types have to be generated. Otherwise, search results may be biased to one intent and lose other intents. This requirement has not been considered in the existing text-based NER. Traditional NER only considers generating the best entity type. This is weakness of all existing methods including the CRF.

The most widely used NER method is CRF which has the state-of-the-art performance in text-based NER, as reported (McCallum and Li, 2003). The open source NER toolkit developed by Stanford University uses CRF as the main algorithm [1]. However, we find that the CRF is not a good method for dealing with query intent ambigurity. Even if CRF generates multiple (top 3) results, those results can be very similar. It is because the CRF scoring method is based on the whole query entities which can be approximated as the sum of

[1] http://nlp.stanford.edu/software/CRF-NER.shtml

	interpretation	subjective judgement
1	[napa]_Place [auto repair]_Business	Excellent
2	[napa]_Place [auto]_Business [repair]_token	Good
3	[napa]_Place [auto]_token [repair]_token	Fair
4	[napa auto repair]_Organization	Excellent
5	[napa]_Organization [auto repair]_Business	Excellent

Table 1: An example of ambiguous query interpretation. Last column is subjective judgement.

every entities' score. The whole query score is very likely to be dominated by a single token. That is, the global score is controlled by a strong local maximum. Due to this, multiple output can have very similar results. Diversity is an inherent problem for the CRF. We will further explain this in section 3. An example is shown in Table 1. For the query "napa auto repair", "napa"'s meaning is ambiguous. It could be a city name or a business name. Table 1 lists five interpretations from the CRF. To simpify, we use "interpretation" to mean one result of whole entity tagging on the query. All of the top three interpretations tag "napa" as Place because it is a dominated entity. The last two labels "napa" as Organization. But the last two can probably be missed if only top three are used.

In light of CRF's diversity problem on query NER, this work adopts a decision tree based method on the ground of the CRF based results. In this framework, the CRF plays the role as one of query interpretation generation methods, together with other two methods: dictionary based and rule-based. All three methods each generates a few candidate interpretations and then the process is followed by a decision tree based reranking. We used gradient boosted decision tree (GBDT)(Friedman, 2002; Ye et al., 2009) model. We found this model could reduce diversity problem by two reasons: one is global features and using editorial subjective judgement as training target. We give the details in section 4.

2 Related Work

NER has been a long time research area in natural language processing. Traditional NER target is text-based named entity. All sort of machine learning methods have been applied this field and there are many publications. Traditionally, methods can be classified as unsupervised (Collins and Singer, 1999; Etzioni et al., 2005), semi-supervised (Riloff and Jones, 1999; Ji and Grishman, 2006) and supervised according to whether human labelled data is used in training. It can be support vector machine (Asahara and Matsumoto, 2003), hidden markov model (Shen et al., 2003), maximum entropy (Chieu and Ng, 2003), conditional random fields (McCallum and Li, 2003) according to algorithm. The best NER result evaluated in the CoNLL conferenece (Tjong Kim Sang and De Meulder, 2003) using NER shared-task data is 88.76% in terms of F1, where the authors used a multiple classifier combining method linear,HMM, transform based learning (TBL), and ME. Recently, deep learning method is found to achieve state-of-the-art results, 89.59%, on the same data (Collobert et al., 2011) .

CRF is widely used in many NLP tasks including word segmentation (Xue and Shen, 2003), parsing (Sha and Pereira, 2003) and named entity recoginition (McCallum and Li, 2003). In all these fields, CRF was found higher accuracy than the existing methods such as SVM, HMM and ME.

There are many existing work in text-related NER as mentioned above. Less literatures have been found for query NER. Actually, query NER is very important for search because 85% of web search queries contains one or more entities (Lin et al., 2012). The work of (Lin et al., 2012; Pantel et al., 2012) describes their approach to find query intent where EM algorithm is applied on large query set and using url-click information. In their work, 73 entity type is defined and 147K entities are extracted from Freesbase. Their work limits only one single entity inside a query.

There are some work on query segmentation. The task is to divide query into multiple segments correctly based on entity but not need to label entity type. (Hagen et al., 2012) described an overview of state-of-the-art work in query segmentation. The best word segmentation results are around 85%.

The paper (Guo et al., 2009) proposes a probabilistic approach to the named entity recognition using query log data and Latent Dirichlet Alloca-

tion. Topic model is constructed by a weakly supervised LDA method. The method is tested on a tag set with only 4 taxonomies related to media. The best NER accuracy as reported is 80% on the four categories. However, the evaluation was based on known query segmentation. The results will be worse if segmentation is not known as in our work.

Decision tree (DT) method was used in the early stage of NLP (Black et al., 1993; Magerman, 1995). This approach was replaced by more advanced models such as maximum entropy and CRF. The old DT is a classification model that mixmizes classification error rate. Our DT model is based on the work of (Friedman, 2002; Ye et al., 2009), gradient boost decision tree (GBDT). The DT in our work is a regression model that optimizes editorial grades. The model has been widely applied into many machine learning tasks, for example, Web search ranking (Ye et al., 2009).

3 Non-diversity of the CRF tagging model

We used three approach methods to generate query interpretation candidates: maximum length match (MLM), heuristic rule based (HRB), and CRF tagging. CRF tagging is the most effective method. But MLM and HRB can provide additional candidates. Our NER system has a knowledge repository that stores millions of entities. Query interpretation by the MLM is through entity match with the entites in the knowledge base. Use the longest query words from left to right to match entites until a match is found. Continue to match anther longest entity starting from the unmatched position of the query. Continue the process until all query words are gone through. This is a greedy search from left to right. MLM is able to generate candidates such as No.1, No.4 and No.5 of Table 1 because it is either full match of an entity or partial match by two entities.

HRB is based on rules. We have created rule base consisting of hundreds of patterns. Each pattern defines a full parse consisting of words and entities. And a score is given for each rule. The score is proportional to frequency of the rule occurring in the query log. For example, No.1 of Table 1 can be generated by "Place + Business" pattern.

The CRF is undirected graphical model (Lafferty et al., 2001). The CRF tagging illustration is shown in Fig. 1. The figure shows a six-word query. The query sequence is $X = x_0, x_1, \cdots, x_N$. The labels are $Y = y_0, y_1, \cdots, y_N$. y_i is x_i's tag. y_i is one tag of a candidate tag set. Figure 1 shows three candidate tags for each word. These candidates are decided by the CRF tagging model that calculates the joint probability of the entire sequence of labels as the following formula,

$$p(Y|X) = \frac{1}{Z(X)} \prod_{t=1}^{T} exp \sum_{k=1}^{K} \theta_k f_k(y_t, y_{t-1}, x_t) \quad (1)$$

$f_k(y_t, y_{t-1}, x_t)$ is features. θ_k is model's parameters, which are estimated by maximizing log likelihood of training data.

Which path has the highest value of will be the top 1 resulting tag sequence. Fig. 1 also illustrates local maximum problem. If $p(y_2) = y_{21}|X) \gg p(y_2 \neq y_{21}|X)$, all tag sequences will pass through y_{21}. As results, y_{22} and y_{23} are unlikely to be chosen. There are multiple reasons to have local maximum. *Label bias* in training is one cause. Training data may be biased to one entity over others. Like query "apple tree". If majority of "apple" in training has "Organization" entity type, it is unlikely to recogize "apple tree" as "Fruit". The label bias is very easy to happen for query NER because the entities are huge and diversified. *Label bias* issue was raised in (Lafferty et al., 2001). While the CRF is a theoratical solution to attack label bias problem (Lafferty et al., 2001), we observed a large amout of local maximum for query NER problem. There are other reasons to cause local maximum like local features. CRF uses many local features instead of global features. Either global features are not easy to implement or it is not efficient in the CRF method. Local features may cause the local maximum. On the contrary, GBDT use a lot of global features.

We adopted the common methods, "BIO" tagging, for NER tagging using the CRF (Xue and Shen, 2003). "B" and "I" specify an entity boundary, either the beginning or intemediate position of an entity. "O" indicates non-entity word. This method can map a query word sequence to an entity sequence throught one-by-one alignment. For example, "tom cruise movie" maps to "B_Person I_Person O". This mapping method is a typical use of the CRF tagging method. Named entities can be easily obtained by simply mapping the BIO se-

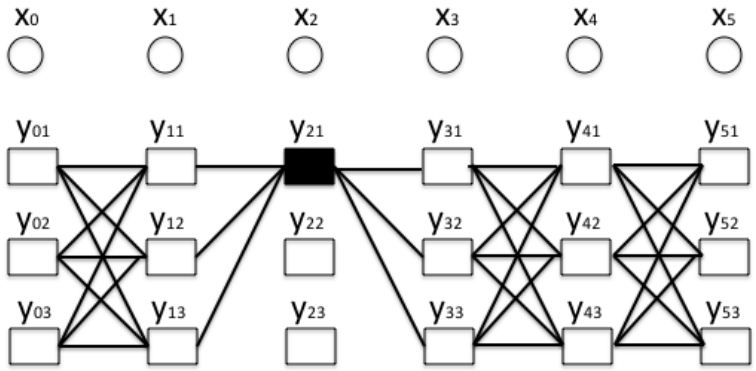

Figure 1: CRF local maximum: y_{21}, y_{22}, y_{23} are word x_2's candidates tags. y_{21} is a local maximum tag. All top tag sequences have to pass it other than two other tags.y_{21} suppresses y_{22} and y_{23} in ambiguous query tagging.

quence to entity sequence. Both entity and entity boundary are extracted.

This work used many new features that is unique for query NER. Our model used about 30M features due to diversified web vocabulary and rich query context. We used the following information as features: precede/current/next query word, word position, word boundary, word spelling, lexical and query word topics. Topics were created by the latent dirichlet allocation (LDA) over 10 million queries. Our training data were selected randomly from query log and manually labeled by editors.

In decoding stage, multiple interpretations can be produced and the order of interpretations is ranked according to the value of $p(Y|X)$. Note $p(Y|X)$ is conditional probability. The best tag sequence has the highest value holds only in condition of given X. $p(Y|X)$ is meaningless globally. We can't judge whether a CRF interpretation is good or not according to $p(Y|X)$. What we can only tell is the best tag sequence relative to other results given the tag sequence,X. Using $p(Y|X)$ as a criteria to select good interpretations was disappointed. However, GBDT can tell good interpretations from bad ones because the GBDT score is meanful globally.

4 GBDT rescoring model

The re-scoring function is a logistic regression model. The probability is given by the formula below,

$$p(x) = \frac{1}{1+e^{(a*(b-f(x))}}$$
$$y = \begin{cases} +1 & p(x) > threshold \\ -1 & otherwise \end{cases} \quad (2)$$

where x is the generated feature vector for a given query and interpretation. a and b are the slope and pivot parameters. $a = 2.0$ and $b = 2.6$ are determined by optimizing classifier's performance in terms of recall and precision. $\{+1, -1\}$ denotes good interpretation and bad interpretation. $thrshld$ is set to separate good and bad interpretation according to the score $p(x)$.

We employ Gradient Boosted Decision Tree algorithm to learn the function $f(x)$. Gradient Boosted Decision Tree is an additive regression algorithm consisting of an ensemble of trees, fitted to current residuals, gradients of the loss function, in a forward step-wise manner. It iteratively fits an additive model as

$$f_t(x) = T_t(x;\Theta) + \lambda \sum_{t=1}^{T} \beta_t T_t(x;\Theta_t)$$

such that certain loss function $L(y_i, f_T(x+i))$ is minimized, where $T_t(x;\Theta_t)$ is a tree at iteration t, weighted by parameter β_t, with a finite number of parameters, Θ_t and λ is the learning rate. At iteration t, tree $T_t(x;\beta)$ is induced to fit the negative gradient by least squares.
The optimal weights of trees β_t are determined by

$$\beta_t = \text{argmin}_\beta \sum_{i}^{N} L(y_i, f_{t-1}(x_i) + \beta T(x_i, \theta))$$

GBDT model is supervised machine learning. It needs manually created training data. The process of creating training data is as follows. We randomly sampled hundreds of thousands of queries from Web search query log. For each query, we

run CRF tagger to get top candidate interpretations. In addition, we generated more interpretations by MLM and HRB. Editors assign 4 grades EGFB (Execllent, Good, Fair, Bad), to each of (query, interpretation) pairs. We show some examples on the last column of Table 1. Grade 'E' is assigned to an excellent interpretation where all entities are correctly spanned and tagged. This interpretation will almost definitely lead to search results that are much better than those generated by the 'default interpretation' (where all words are just 'token'). Grade 'G' is assigned to a good interpretation where some entities but not all are correctly spanned and tagged. For example, some entities will not be tagged at all or tagged with a very general tag. This interpretation may lead to search results that are better than the default interpretation, but these could be further improved. Grade 'F' is the baseline grade and default level of interpretation. It is no better or worse than if each word was treated as an individual keyword match. Queries with all spans marked as "token" are considered Fair. The exception to this rule is a query whose only interpretation are all spans labeled "token". In this case, the interpretation is marked Good. Grade 'B' is a bad interpretation; some of the important entities are not spanned or tagged correctly, in a way that leads to a wrong understanding of the query.

5 Features of GBDT

Features used in the GBDT are completely different from those in the CRF. GBDT feature value can't be string. Contextual features can't be used in GBDT. All GBDT features' values are real values. Feature size is very different. We used 30M features in CRF, but only 500 in GBDT. But GBDT can use global features. We show some global features in Table. 2. Those global features are not available in CRF. Some features are very useful. For example, No.1 is number of recognized entities. It is a good feature to distinguish "napa" as Place from as Organization. is No.3 is tag language model (LM) score for whole sentence. No.4 is count of generation methods that can generate the interpretation. Its function likes a voting strategy. No.6 is number of entities stored in taxonomy.

GBDT features are generated by the following methods. Eight features are from query level. (1) number of terms (2) number of stop words (3) number of special characters (4) number of entities recognized for this interpretation (5) number of non-tagged terms (6) number of non-token tags (7) position of first non-token tag (8) position of last non-token tag.

1	number of entities
2	number of non-recognized tokens
3	tag LM score of whole query
4	number of generation methods
5	crf model score
6	number of taxonomy match

Table 2: GBDT global feature examples

Four features are from CRF tagger: (1) CRF raw probability (2) rank of CRF generated interpretation (3) ratio of current CRF interpretation to the top 1 CRF interpretation. (4) a boolean feature to indicate if interpretation is from CRF tagger.

There are two types of LM scores, context-independent tag (priors) and context dependent tag ngram LM scores. Context-independent tag scores are the average prior scores of any tags in the interpretation. Context-dependent tag sequence scores are ngram LM probability. Context-independent scores do not count token but context-dependent scores include tokens. LM was built using editorial judgment data. 200 features are extracted by this method.

Some other features include template match. We manually wrote a lot of template-based rules to recognize query intents. These rules are from multiple domains. Take the query above as examples. (napa, Place) matches a Local template rule and (napa, Organization) matches a Auto template rule. Features from template match include number of rule match.

Other features include counts of each taxonomy and ratio of each taxonomy if the query interpretation has multiple taxonomies. The feature size of this type is equal to the tag set.

We defined entity confidence score based on CRF results. It is called global word posterior probability in (Soong et al., 2005). Assume CRF generates N query interpretations. Entity e may or may not have appearance in a interpretation I_i. If the CRF score of the interpretation is as $P(I_i)$, then the confidence of entity is calcuated by the formula as,

$$\text{conf}(e) = \frac{\Sigma_i^N P(I_i)\epsilon(e, I_i)}{\Sigma_i^N P(I_i)}$$

Above is confidence score of one entity. From it, we genereate full interpretation score as a new feature. It is calculated as sum of all entities' confidence score recognized in the interpretation.

6 Experiments

We predefined an in-house taxonomy set. This set has two levels of granularity: coarse level and fine level. Coarse level has 20 entity categories and Fine level has about 200 categories. Most of the widely defined categories are included in our taxonomy set such as Local, Organization, Person, Product. There are also some new categories defined specifically for business purpose. Besides type Organization, we define Brand category to emphasize shopping intent. For instance, "apple" is an Organization for query "apple store", and BRAND for "apple iphone". The differing categories makes clear to separate Local intent and Shopping intent. "apple store" has Local intent that wants to show store location, driving direction, or store hours. "apple iphone" has Shopping intent that shows iphone type or price list.

For each Coarse category, a few Fine categories are defined. As an example, we define Actor, Director, Politician for Person category. Similarly, each coase category is assoicated some fine category.

Both CRF and GBDT are supervised machine learning methods. CRF needs editors to label entity for a given query. GBDT needs editors to judge quality grade for a given query interpretation according to 4-grade judgement guideline. We submitted 60K queries to editors and returned about 80K perfect interpretations which are used for CRF training. We collected 200K query interpretations from output of MLM,HRB and CRF and asked editors to judge these interpretations. We took 5K interpretations from 200K as evaluation data set. These 5K interpretations contain 600 queries and the average interpretations per queries is 8.2. It has 955 Excellent grade, 1051 Good, 974 Fair and 1960 Bad.

Some ambiguous queries are shown in Table 6. The examples show ambigurity could happen possiblly in many cases such as between Brand and Product (flash player), Business and Organization (zurich life insurance) and Media and Person (billy gilman).

The evaluation is based on the Accuracy-N and the Discounted Cumulative Gain (DCG) (Jarvelin and Kekalainen, 2000). Accuracy-N is calculated as "the number of interpretations that are perfect on top N interpretations divided by the total number of interpretations on top N". Each query has multiple perfect interpretation.

The DCG is computed as follows:

$$\text{DCG-N} = \sum_{i=1}^{N} \frac{g(i)}{log(1+i)}$$

where $g(i)$ is the gain associated with the rating of result at rank i and N is the maximum depth result to consider. In this paper, we use gains of $7, 3, 0.5, 0$, respectively, corresponding to the four ratings or relevance grades.

Table 3 shows the comparison between CRF tagger and GBDT. Both coarse level and fine level results are presented. We list top 3 results. We observed the CRF tagger is better than GBDT if top 1 interpretation is considered. But, in case of more than one interpretations, GBDT is much better than CRF. The reasons to have improved multiple interpretations are three folds: global features, subjective training targets and using MLM and HRB to increase CRF candidates. MLM and HRB provides more diversified interpretations. Actually, accuracy of MLM and HLB are much lower than CRF. On the same evaluation set, MLM's Accuracy-1 is 0.40 and HRB's Accuracy-1 is 0.53. Both have much lower accuracy than CRF, 0.72. Even though, GBDT can take all the three methods' benefits.

The second test to compare CRF tagger and Scorer is to evaluate precision and recall of good interpretations. The test set is (query, interpretation) pairs evaluated with EGFB grades. The CRF tagger and GBDT assign scores to each (query, interpretation) pair. A threshold can be set to determine the boundary of good and bad interpretation according to the score. Precision is calculated by the number of correctly detected E and G interpretations divided by the total number of recognized E/G interpretations. Recall is calculated by the number of correctly detected E and G interpretations divided by the total number of E and G interpretations of answers. The precision recall graph is shown in Fig. 2. Each point is the precision and recall values corresponding to a threshold. The top green curve is of GBDT and the bottom blue curve is CRF tagger. We see Scorer is much better than CRF tagger in terms of both precision and recall.

		Coarse Category		Fine Category	
		Accuracy-N	DCG-N	Accuracy-N	DCG-N
N=1	CRF	0.7249	5.3	0.5987	4.5
	GBDT	0.6814	5.2	5647	4.1
N=2	CRF	0.5151	7.9	0.4187	6.5
	GBDT	0.5324	8.1	0.4528	7.4
N=3	CRF	0.3794	9.2	0.3325	7.1
	GBDT	0.4203	9.5	0.3972	8.5

Table 3: Comparison of CRF and GBDT

Query	*Interpretation*
adobe flash player adobe flash player	[adobe]_Brand [flash player]_Brand [adobe]_Brand [flash player]_Product
zurich life insurance zurich life insurance	[zurich]_Organization [life insurance]_Business [zurich life insurance]_Organization
cars cars	[cars]_Media:Movie [cars]_ Product:Auto
windsor hotel windsor hotel	[windsor]_Place [hotel]_Business [windsor hotel]_Organization
albany rv albany rv	[albany]_Place [rv]_Product [albany rv]_Organization
billy gilman billy gilman	[billy gilman]_Media:album [billy gilman]_Person

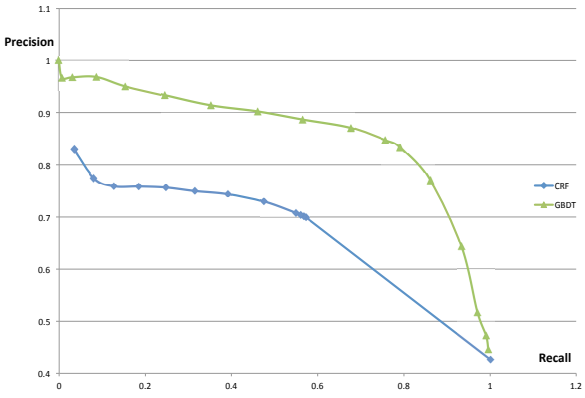

Figure 2: Comparison of CRF and GBDT

7 Conclusions

Web query NER poses more challenge than text NER because the query is ambiguous and short of contextual information to disambigurate. Therefore, diversified results become new problems for query NER. Our work find the flaw of the common CRF based NER when it is applied to query NER. We prove the CRF NER can't be solely used because of its local maximum, non-diversity, low precisions, and high false positive. This work proposed a rescoring GBDT approach to fix the CRF's problem. Diversity problems are significantly reduced by applying multiple approaches: global features, global subjective judgement and addition of MLM and HRB with the CRF candidates. Our experiments prove this new framework can achieve our goals. Both precision and DCGs are improved over the CRF method. Acturally, The CRF and GBDT function differently. The CRF tagger can generate the most likely interpretations because it optimizes the likelihood of the training data. GBDT is a regression model that maximizes subjective judgements. It can make better judgement in classifying good and bad interpretations. This is exactly what query NER wants.

References

Masayuki Asahara and Yuji Matsumoto. 2003. Japanese named entity extraction with redundant morphological analysis. In *Proceedings of the 2003 Conference of the North American Chapter of the Association for Computational Linguistics on Human Language Technology - Volume 1*, NAACL '03, pages 8–15, Stroudsburg, PA, USA. Association for Computational Linguistics.

Ezra W. Black, Roger Garside, and Geoffrey N. Leech, 1993. *Statistically-driven computer grammars of*

English: The IBM/Lancaster approach. Vol. 8. Rodopi.

Hai Leong Chieu and Hwee Tou Ng. 2003. Named entity recognition with a maximum entropy approach. In *Proceedings of the Seventh Conference on Natural Language Learning at HLT-NAACL 2003 - Volume 4*, CONLL '03, pages 160–163, Stroudsburg, PA, USA. Association for Computational Linguistics.

Michael Collins and Yoram Singer. 1999. Unsupervised models for named entity classification. In *Joint SIGDAT Conference on Empirical Methods in Natural Language Processing and Very Large Corpora*.

R. Collobert, J. Weston, L. Bottou, M. Karlen, K. Kavukcuoglu, and P. Kuksa. 2011. Natural language processing (almost) from scratch. *Journal of Machine Learning Research*, 12:2493–2537.

Oren Etzioni, Michael Cafarella, Doug Downey, Ana-Maria Popescu, Tal Shaked, Stephen Soderland, Daniel S. Weld, and Alexander Yates. 2005. Unsupervised named-entity extraction from the web: An experimental study. *Artif. Intell.*, 165(1):91–134, June.

Jerome H. Friedman. 2002. Stochastic gradient boosting. *Comput. Stat. Data Anal.*, 38(4):367–378, February.

Jiafeng Guo, Gu Xu, Xueqi Cheng, and Hang Li. 2009. Named entity recognition in query. In *Proceedings of the 32nd international ACM SIGIR conference on Research and development in information retrieval*, SIGIR '09, pages 267–274, New York, NY, USA. ACM.

Matthias Hagen, Martin Potthast, Anna Beyer, and Benno Stein. 2012. Towards optimum query segmentation: in doubt without. In *Proceedings of the 21st ACM international conference on Information and knowledge management*, CIKM '12, pages 1015–1024, New York, NY, USA. ACM.

Kalervo Jarvelin and Jaana Kekalainen. 2000. IR evaluation methods for retrieving highly relevant documents. In *SIGIR*, pages 41–48.

Heng Ji and Ralph Grishman. 2006. Data selection in semi-supervised learning for name tagging. In *Proceedings of the Workshop on Information Extraction Beyond The Document*, IEBeyondDoc '06, pages 48–55, Stroudsburg, PA, USA. Association for Computational Linguistics.

John D. Lafferty, Andrew McCallum, and Fernando C. N. Pereira. 2001. Conditional random fields: Probabilistic models for segmenting and labeling sequence data. In *Proceedings of the Eighteenth International Conference on Machine Learning*, ICML '01, pages 282–289, San Francisco, CA, USA. Morgan Kaufmann Publishers Inc.

Thomas Lin, Patrick Pantel, Michael Gamon, Anitha Kannan, and Ariel Fuxman. 2012. Active objects: actions for entity-centric search. In *Proceedings of the 21st international conference on World Wide Web*, WWW '12, pages 589–598, New York, NY, USA. ACM.

David M. Magerman. 1995. Statistical decision-tree models for parsing. In *In Proceedings of the 33rd Annual Meeting of the Association for Computational Linguistics*, pages 276–283.

Andrew McCallum and Wei Li. 2003. 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. Association for Computational Linguistics.

Tomas Mikolov and Geoffrey Zweig. 2012. Context dependent recurrent neural network language model. In *Spoken Language Technologies*. IEEE.

Patrick Pantel, Thomas Lin, and Michael Gamon. 2012. Mining entity types from query logs via user intent modeling. In *Proceedings of the 50th Annual Meeting of the Association for Computational Linguistics: Long Papers - Volume 1*, ACL '12, pages 563–571, Stroudsburg, PA, USA. Association for Computational Linguistics.

Ellen Riloff and Rosie Jones. 1999. Learning dictionaries for information extraction by multi-level bootstrapping. In *Proceedings of the Sixteenth National Conference on Artificial Intelligence and the Eleventh Innovative Applications of Artificial Intelligence Conference Innovative Applications of Artificial Intelligence*, AAAI '99/IAAI '99, pages 474–479, Menlo Park, CA, USA. American Association for Artificial Intelligence.

Fei Sha and Fernando Pereira. 2003. Shallow parsing with conditional random fields. In *Proceedings of the 2003 Conference of the North American Chapter of the Association for Computational Linguistics on Human Language Technology - Volume 1*, NAACL '03, pages 134–141, Stroudsburg, PA, USA. Association for Computational Linguistics.

Dan Shen, Jie Zhang, Guodong Zhou, Jian Su, and Chew-Lim Tan, 2003. *Proceedings of the ACL 2003 Workshop on Natural Language Processing in Biomedicine*, chapter Effective Adaptation of Hidden Markov Model-based Named Entity Recognizer for Biomedical Domain.

Frank K. Soong, Wai kit Lo, and Satoshi Nakamura. 2005. Generalized word posterior probability (gwpp) for measuring reliability of recognized words.

Erik F. Tjong Kim Sang and Fien De Meulder. 2003. Introduction to the conll-2003 shared task:

Language-independent named entity recognition. In *Proceedings of the Seventh Conference on Natural Language Learning at HLT-NAACL 2003 - Volume 4*, CONLL '03, pages 142–147, Stroudsburg, PA, USA. Association for Computational Linguistics.

Nianwen Xue and Libin Shen. 2003. Chinese word segmentation as lmr tagging. In *Proceedings of the Second SIGHAN Workshop on Chinese Language Processing - Volume 17*, SIGHAN '03, pages 176–179, Stroudsburg, PA, USA. Association for Computational Linguistics.

Jerry Ye, Jyh-Herng Chow, Jiang Chen, and Zhaohui Zheng. 2009. Stochastic gradient boosted distributed decision trees. In *Proceedings of the 18th ACM conference on Information and knowledge management*, CIKM '09, pages 2061–2064, New York, NY, USA. ACM.