Journal of Data Intelligence and Predictive Systems (JDIPS) **2025, 1(1): 1-7**
https://www.longmanpublishers.com/

PARTICULAR ANNOTATION SCHEME FOR ACTIVE LEARNING ON RECOGNITION TASKS FOR NAMED ENTITIES

J Muddathir
Federal University of Applied Sciences Kachia, Kachia, southern Kaduna, Nigeria

Abstract: One interesting method for reducing the high annotation costs associated with creating training data for named entity recognition (NER) tasks is active learning. The efficiency of the data instance selection is constrained, nonetheless, because current active learning techniques on NER tasks implicitly assume the whole annotation scheme, where the entire phrase serves as the unit of an annotation request. In this research, we offer a novel partial annotation scheme-based active learning approach that asks human annotators to identify a specific portion of the target phrases after choosing a portion of the sentences to be annotated. In contrast to the current active learning techniques on NER problems, we demonstrate in the experiment that the partial annotation strategy can train the suggested point-wise prediction model rapidly.
Keywords: Neural networks, Named entity recognition, Text tagging.

1. Introduction

One of the core procedures in natural language processing is named entity recognition (NER), which automatically extracts named entities (such as a person's, organization's, or location's name). Information retrieval and question answering are two examples of natural language applications that benefit from NER's fundamental semantic awareness [1]. In recent years, there has been an increase in the amount of digitalized text content available in a wide range of specialized sectors, including patents, recipes, discussions in programming forums, papers in a particular field of study, etc. Because specialized domains have their own vocabulary that conventional NER models are unable to identify, there is a growing need to develop a specific NER model for processing named entities in each domain appropriately.

The expense of creating training data, sometimes referred to as an annotation corpus, is one of the main problems while training a bespoke NER model. A set of sentences with named entity tag annotations on the phrases within the sentences makes up the NER annotation corpus. Domain experts must supply these annotations using an annotation interface (Fig. 1 (Left)). The cost of creating training data for NER tasks is high because the annotation process necessitates domain knowledge and time-consuming labor to read the entire phrase in order to identify all the identified entities.

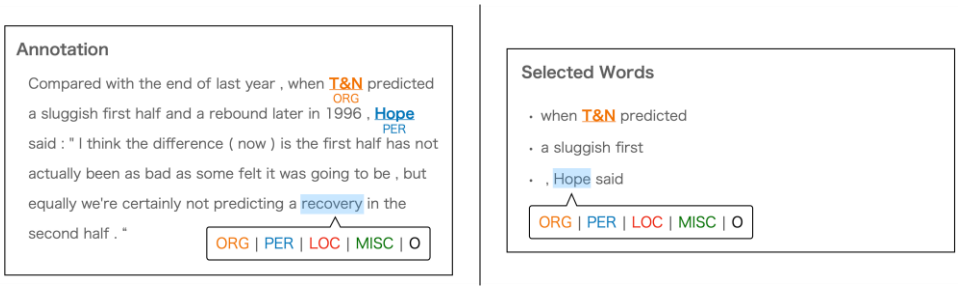

Fig. 1. (Left) In annotation task for NER in the full annotation scheme, human annotators are asked to check a whole sentence. **(Right)** In the partial annotation scheme, human annotators are asked to check a specific short part in sentences.

Active learning is one of the promising strategies to lower the annotation cost. A technique known as "active learning" looks at how a machine learning model is acting right now and selects a data instance that is thought to be the most helpful for training the model [2]. A machine learning model's performance can be considerably enhanced by manipulating the order of the sentences' annotation requests, as opposed to training the model with the same number of sentences annotated in a random order. Up to now, active learning techniques for NER models have been put forth [3, 4]. These techniques, however, have a potential drawback in that they might lose out on a chance to perform better because they make an implicit assumption about the annotation scheme. The current active learning techniques are suggested under a full annotation scheme, meaning that the algorithm

selects a sentence and requests that human annotators review the entire sentence. The granularity of data instance that an active learning system can select is coarse, particularly when the sentence is lengthy. We must ask human annotators to review the entire sentence for phrases that the current NER model may already identify, even if it only needs to learn a specific phrase in a sentence.

We examine a partial annotation strategy that asks human annotators to review a brief portion of the target phrases at each request in order to solve this possible problem (Fig. 1 (Right)). In contrast, we suggest a novel active learning approach that chooses a portion of the phrases to be annotated. In comparison to active learning on the entire annotation scheme, it is anticipated that the fine-grained annotation request will result in a cheaper annotation cost for obtaining the same performance. Reducing the restriction on the training data's structure is the difficult part of putting this plan into practice. Actually, the main NER models assume that a phrase is the unit of training data instance. That is, we cannot use phrases that are only partially annotated to train such models. By implementing a point-wise prediction model [5] that can be trained using partially annotated phrases, the suggested approach in this article circumvents this issue.
The structure of this document is as follows. Research on the connection between NER and active learning is included in Section 2, along with a few articles on the use of active learning for the named entity recognition task in particular. We outline the point-wise prediction approach for named entity extraction and its application to active learning in Section 3. Following an experiment to demonstrate the efficacy of active learning for point-wise prediction, we describe a comparison experiment between the current approach and our method in Sections 4 and 5. We go over the findings and next steps in Section 6.

Sentence: x	Cuban	novelist	Jose	Soler	Puig	dies	at	79	.
Labels: y	B-MISC	O	B-PER	I-PER	I-PER	O	O	O	O

(a) Full annotation corpus

Sentence: x	Cuban	novelist	Jose	Soler	Puig	dies	at	79	.
Labels: y	-	O	B-PER	I-PER	I-PER	-	-	O	O

(b) Partial annotation corpus

Fig. 2. Example of annotation corpus

2. Related Work

2.1. Recognition of named entities

A sequence labeling task that predicts the labels for each word is how NER challenges are usually constructed. Several tag formats (such as IOB2, IOE2, and IOBES) have been proposed to represent the information specifying named entities. We use the IOB2 format in this work. The IOB2 format uses an entity type tag with the prefixes "I," "B," or "O" to represent each word's role. "B" stands for "Begi," which is the named entity phrase's initial word. "I" stands for "Inside," which is the second or subsequent word in the phrase "named entity." "O" denotes "Other," which means the word does not appear in named entity phrases. The sequence labels in the IOB2 format with two entities—"Jose Soler Puig" as a person's name (PER) and "Cuban" as a miscellaneous entity name (MISC)—are demonstrated in Fig. 2(a). Every job or domain has a defined entity type, such as PER or MISC.

The two types of annotated corpora that we examine in this paper are complete annotation corpora and partial annotation corpora. Every word x in a complete annotation corpus has the label y assigned to it (see Fig. 2(a)). On the other hand, just a subset of words are tagged in a partial annotation corpus (Fig. 2(b)). The well-known CoNLL-2003 dataset [6] is a fully annotated corpus for NER in a general domain. It comprises news items annotated with four different entity types: person, organization, location, and miscellaneous named entity.

With the CoNLL-2003 corpus, Florian et al. [7] obtained a high performance of 88.76% in F1 value by combining several machine learning models. McCallum et al. [8] suggested a named entity recognition technique that makes use of conditional random fields (CRF) after considering named entity recognition as a sequence labeling task. A technique that employs convolutional neural networks (CNN) for word sequences was recently proposed by Ronan et al. [9]. Deep learning-based named entity recognition algorithms have since gained popularity. A model by Huang et al. [10] replaces Ronan's CNN encoder with bidirectional long short-term memory (LSTM). Bidirectional LSTM was used by Lample et al. [11] to represent word-level and character-

level information. Using the CoNLL-2003 corpus, Ma et al. [12] created a model that combines bidirectional LSTM with a CNN and CRF, and it obtained an F1 value of 91.21%. This model performed better without pretreatment of manually prepared data since it leverages CNN for character-level information. These techniques, however, are unable to employ a partial annotation corpus because they are designed to build a named entity recognition model utilizing a full annotation corpus as training data. Annotators must therefore label every word in a phrase [13]. Although there have been proposals for CRF that employ margin likelihood to use a partial annotation corpus, the training period is somewhat lengthy. Because active learning necessitates recurrent learning, it is therefore impractical to apply it to these models [14].

2.2. Learning that is active

One technique for increasing accuracy while requiring less annotation work is active learning. A machine learning model examines an unlabeled dataset in active learning and chooses a data instance that appears to be the most instructive for the machine learning model at that moment. An oracle (usually an annotator) who is aware of the answer annotates the chosen instance with its true label. Asking the oracle is referred to as a "query" or "request." The oracle-labeled cases are appended to the training dataset and used to retrain the model. This procedure is repeated to gather useful labeled data. Although there are a number of active learning situations [2], pool-based active learning is used in this work. When a lot of unlabeled data is gathered and stored, pool-based active learning chooses a data instance for model training. In situations like pool-based active learning, each instance's information metrics are evaluated before a query is made. For calculating information metrics, several query strategies have been put forth.

One technique for determining the label of the data instance with the most uncertain prediction based on the current model is uncertainty sampling [15]. The uncertainty labels can be chosen in a number of ways. First, if x is the sentence and $y*$ is the label with the highest posterior probability, this is the simplest and least confident approach.

The amount of information considered is greater than that of the least confident method because margin sampling takes into account ambiguities other than the most likely label sequence.

Query by committee (QBC) [16] is another query approach that makes use of several models of $C = \{\theta(1),..., \theta(C)\}$. Each instance's label is predicted by the model-based commission, which also takes into account the cases where a variety of prediction labels provide the most valuable information. In order to gauge the extent of the disparity, QBC additionally recommends measurements such vote entropy [16] and the Kullback-Leibler divergence [17]. Other approaches have been put forth, such as expected model change [18], which queries the data instance that is anticipated to have the biggest influence on the parameter update, and expected error reduction [19], which queries the instance that minimizes generalization errors.

2.3. Effective NER model training

Because it involves expert knowledge and annotation, creating training data for NER models is costly. As a result, several approaches have been put forth to effectively train NER models. To lessen the work required to annotate the position of phrases that signal named entities, Patra et al. [20] created a NER model that can be trained using simply information on whether a sentence contains named entities. To create a NER model that combines annotations from several workers and enables training without an expert annotator, Nguyen et al. [21] used crowdsourcing. In order to make high-performing predictions even in the absence of training data in the domain to be annotated, Lison et al. [22] created a model that can combine predictions from several NER models trained in other domains. Active learning has also been applied to NER in a number of research.

In order to solve the sequence labeling task, Settles et al. [3] suggested using active learning with conditional random fields. When many named entities need to be extracted from deep learning models, Yanyao et al. [4] have demonstrated that CNN training is quicker than LSTM training. They suggested a CNN-CNN-LSTM model, which consists of an LSTM, a word CNN, and a character CNN. However, annotators must identify every word in a phrase because this model cannot be trained with a partial annotation corpus. As a result, annotators must mark phrases that are useless for model training, which raises the expense. In this research, we apply active learning to a point-wise prediction model that can be trained using a partial annotation corpus in order to lower annotation costs in the partial annotation scheme.

3. Method

3.1. Point-wise prediction for named entity recognition

In this study, we expand the Neubig et al. [23] point-wise prediction method to named entity extraction. Point-wise prediction models are explained in this section. Let y = ▨ y1, y2,..., yT ▩ be a corresponding label (tag) sequence and x = ▨ x1, x2,..., xT ▩ be a phrase. Point-wise prediction models treat this task as a multi-class classification issue for each yi, rather than taking into account the likelihood of the label sequence as in conditional random fields. To address multi-class categorization, machine learning techniques such logistic regression models, support vector machines, and decision trees were created.
For point-wise prediction in this work, we use a multi-class logistic regression model. When Y = {1, 2,..., K}, the multi-class logistic regression model predicts a single label y ∈ Y.

The gradient descent approach is an efficient way to determine the weight vector w since the likelihood is a convex function.

We define window width m to extract features on point-wise prediction. Features taken from the surrounding words xi−m,..., xi+m are used to predict the label yi at location i in a sentence (see Fig. 3). Our method rewrites multi-class logistic regression to produce the following equation.

Using "feature templates," we define the feature functions f = f1,..., fK. In the field of Japanese word segmentation, Neubig et al. [23] established feature templates for point-wise prediction. We adapt them to the English NER domain. The following feature templates are specified in the suggested approach.

• Surface: Characteristics derived from one-hot encoding of the (position, vocabulary) pairs that occur in xi−m,..., xi,..., xi+m.
• Word type: xi−m,..., xi,..., xi+m are utilized as features to derive information on the word type. As a feature derived from the word type, we use whether the term starts with a capital letter or is composed entirely of uppercase characters in this work.

One-hot encoding of the pairings of (position, part of speech) that occur in xi−m,..., xi,..., xi+m is used to generate the features for part of speech.

Furthermore, the Mori et al. [5] method uses the one-to-many method (one-versus-rest) to estimate the part of speech after developing a classifier for each possible part of speech of each word. Nevertheless, our approach uses a single classifier to retrieve named things.

Point-wise prediction was employed by Mori et al. [5] for problems involving morphological analysis. Throughout the morphological analysis exercises, nouns appeared multiple times. Consequently, words that were not found in the dictionary or learning corpus might be considered nouns. However, in named entity identification, the initial test shown that when the unknown word was assigned the "O" tag along with another specified tag, the extraction performance significantly declined. For all words, we thus employ a single multivalued classifier.

3.2. Utilizing active education

In order to use the partial annotation scheme when applying active learning with the point-wise prediction model, point-wise prediction can be trained using a partial annotation corpus. Pool-based active learning is the method of active learning employed in this study. The active learning technique known as pool-based active learning makes the assumption that there is a big pool of unlabeled data (U) and a limited amount of labeled data (L). The data instance that appears to be the most helpful for training is chosen from the pool and the label is requested of the annotators in the pool-based approach (which is based on the current model's prediction). The model is updated in accordance with the annotators' labeling of the instance. These procedures are repeated to train the model. A technique known as "sequential active learning" modifies the model every time we get a new tagged instance. While the model is learning, this approach makes the annotator wait. The suggested approach uses "batch active learning," which chooses and searches several unlabeled data instances at once, to cut down on waiting time.

The most common representation of pool-based active learning is Algorithm 1. Initially, the approach uses the function train(·) for the labeled data L to train the model. Then, using the query technique ϕ(·) outlined in Section 2, label queries are sent to the annotator B times from the pool U. The instance of the searched data is taken out of U and added to L as labeled data. In this loop, the model is trained repeatedly. When B > 1, Algorithm 1 is considered batch active learning; when B = 1, it is considered sequential active learning.

Training sentences that are only partially annotated cannot be used to train existing models like CRF and LSTM-

CNN-CRF until the complete sentence is tagged. As a result, when active learning is used, it is necessary to query for the labels of every word in the phrase, as illustrated in Figure 4. However, as illustrated in Fig. 5, when active learning is used for point-wise prediction, parts of the sentence's terms can be independently probed. When the same amount of words are annotated in this manner, active learning using point-wise prediction models can query more significant terms than the current approaches. Thus, it is anticipated that the suggested approach would be trained more effectively.

Here, we describe the suggested method's annotation structure. Annotators must examine the complete phrase in order to decide whether to affix the "B" or "I" tags. Therefore, we examine a technique that displays many surrounding words to annotate the range of the entire named thing. The following is the process for this "word-by-word query": Initially, the query approach is used to choose the word xi. The entire named entity can then be annotated by querying the annotator for an instance of xi−1 if xi is in the middle of the named entity. This query is repeated until the named entity's beginning is visible. This function queries xi+1, the subsequent occurrence in the sequence, until the end of the named entity appears if xi is at the beginning or in the middle of the named entity (see Fig. 6).

4. Experiment

4.1. The dataset

The CoNLL-2003 English [6] named entity dataset was used in this investigation. Two test sets and a training set are included with CoNLL-2003. Each set was separated into training, validation, and test data in accordance with the CoNLL-2003 competition's regulations. A small amount of labeled data (L) and a vast number of unlabeled data pool (U) comprised the training data used to train the model. Only a tiny quantity of labeled data is used in initial training. The deep learning model's hyperparameters were searched using the validation dataset. We assessed the named entity recognition's performance for the test data in the evaluation.

4.2. The experimental procedure

To investigate how well point-wise prediction models work with active learning in the partial annotation scheme, we ran two experiments. In the first experiment, we used point-wise prediction to confirm that active learning works for NER. For point-wise prediction, we contrasted the least confident query approach with the random query technique.

We contrasted the suggested approach with current approaches in the second experiment. As established techniques, we used the LSTM-CNN-CRF named entity recognition model put forth by Ma et al. [12] and the conditional random field. By using F1 values, we evaluated these approaches' performance against that of the suggested approach, the point-wise prediction model. In terms of the LSTM-CNN-CRF hyperparameters, we set the training mini-batch size at 10 and the epoch count at 50.

Uncertain sampling was our query approach of choice. Equation (1) describes the least confidence used by the conditional random field and the point-wise prediction model. We employed the margin sampling [24] outlined in Eq. (2) for the LSTM-CNN-CRF model (for which it is challenging to compute the probability of the entire sequence). We set up batch-type active learning in all techniques, querying roughly 5,000 words at once. Out of all the terms in the unlabeled data, 5,000 words were queried using the query approach for the point-wise prediction. However, neither the LSTM-CNN-CRF model nor the conditional random field were able to classify words separately. As a result, labels were appended to every sentence, and the batch was stopped when there were more than 5,000 tagged words.

We used a corpus whose correct answer was known without manual annotation to conduct active learning in simulated settings. As a result, words that were queried were always assigned the appropriate label. We define a correct named entity extraction in the evaluation as an extracted phrase that precisely fits the named entity's start and end positions as given in the ground truth (i.e., CoNLL-2003 dataset). The accurate extraction is not determined by the number of "O" tags that match the ground truth. For instance, the named object "Jose Soler Pulg" in Fig. 7 was correctly extracted since it matched "PER." The extraction for this named item failed since the model anticipated "O" for the term "Cuban," which had "B-MISC" in the ground truth. As a result, the F1 value is 66.7%, the accuracy is 100%, and the recall is 50%.

5. The outcome

5.1. Making point-wise predictions using active learning

Point-wise prediction models with and without active learning were compared. The words are chosen and queried in a random order in the point-wise prediction model that does not use active learning. The correlation between the amount of words annotated by the simulated annotator and the extraction performance is displayed in Fig. 8. With just 10% of the training data pool, the active learning model achieved the best extraction performance. Thus, using active learning for point-wise prediction lessens the strain on the annotator by requiring less labeled data.

5.2. Using active learning to compare named entity recognition performance

When active learning was used with both the suggested approach and the current methods, Fig. 9 illustrates the correlation between F1 values and the quantity of annotated words. When there was little labeled data, the suggested method outperformed the current strategy in terms of extraction performance.

When there were more annotated words than roughly 40% of the training data pool, other approaches performed better than the suggested strategy. This is the drawback of the suggested approach, which results from the point-wise prediction model's simpler architecture as compared to sentence-wide NER models. Margin sampling was only applied to the LSTM-CNN-CRF model. The performance might be even better if we use margin sampling for point-wise prediction.

5.2.1. A thorough assessment of the suggested approach's and CRF's performance

A more thorough comparison of the suggested approach with the CRF model, which is typical of sentence-wide sequence labeling techniques, is next given in this chapter. We start by looking at each model's named entity labels' F1 scores.

The four named entity types in the CoNLL-2003 dataset are "PER" for a person's name, "LOC" for a place's name, "ORG" for an organization's name, and "MISC," which is a tag applied to the other named entities. The association between the number of words annotated and the F1 score for each named entity type using the CRF and the suggested technique is depicted in Figs. 10(a) and 10(b), respectively. When comparing these two figures, the suggested technique and CRF do not significantly differ in the prediction performance of PER, ORG, and MISC tags. But compared to CRF, the suggested approach for LOC tags improves performance substantially more quickly. The reason for this is most likely because the LOC tags typically begin with a capital letter and that the same named entity appears more often than other tags.

We then look into each model's specific evaluation criteria, including recall, precision, and F1 score. The correlation between the amount of annotated words and the recall, precision, and F1 scores of the CRF and the suggested approach is displayed in Figs. 11(a) and 11(b). Regardless of the quantity of annotated words, the recall in the CRF results is always higher than the precision. At first, there was a significant gap between recall and precision, but as learning progressed, the gap narrowed. On the other hand, the precision is higher than the recall for the suggested approach. For the suggested approach, the precision and recall differences were about equal, but they grew wider as learning went on. The model is not impacted by the very common tag transition of "O" to "O" tag transition, i.e., it does not use the property that words that are not named entities are likely to continue, because the suggested method does not account for the transition of the predicted tag results.

6. Conclusion

Using the partial annotation scheme, we suggested a novel active learning technique that invites human annotators to classify a particular phrase segment. The point-wise prediction model, which may be trained using partially annotated texts, is used in the suggested approach. According to the experimental findings, the suggested approach can be learnt faster than the current active learning techniques that rely on the complete annotation scheme. When we need to quickly train a bespoke NER model on the target domain and have a restricted budget for training data development, this characteristic comes in handy.

The suggested method's performance upper limit is a result of the point-wise prediction model's straightforward architecture. The neural network-based NER models perform better than the point-wise prediction model after we have a significant number of tagged phrases. Overcoming this constraint is a topic we ought to tackle in subsequent study. One such concept is to bridge the gap of the annotation data structure in order to transition from the suggested active learning approach to neural network models. Creating a neural network model that can be trained using a partial annotation corpus is an additional option.

References

1. Diego Mollá, Menno van Zaanen, and Daniel Smith. Named entity recognition for question an- swering. In

Proceedings of the Australasian Language Technology Workshop 2006, pages 51–58, 2006.

2. Burr Settles. Active learning literature survey. Computer Sciences Technical Report 1648, Uni- versity of Wisconsin–Madison, 2009.
3. Burr Settles and Mark Craven. An analysis of active learning strategies for sequence labeling tasks. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, pages 1070–1079, 2008.
4. Yanyao Shen, Hyokun Yun, Zachary C. Lipton, Yakov Kronrod, and Animashree Anandkumar. Deep active learning for named entity recognition. *CoRR*, abs/1707.05928, 2017.
5. Shinsuke Mori, Yosuke Nakata, Graham Neubig, and Tatsuya Kawahara. Morphological analysis with pointwise predictors. *Journal of Natural Language Processing*, 18(4):367–381, 2011.
6. Erik F. Tjong Kim Sang and Fien De Meulder. 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*, pages 142–147, 2003.
7. Radu Florian, Abe Ittycheriah, Hongyan Jing, and Tong Zhang. Named entity recognition through classifier combination. In *Proceedings of the seventh conference on Natural language learning at HLT-NAACL 2003- Volume 4*, pages 168–171, 2003.
8. 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*, pages 188–191, 2003.
9. Ronan Collobert, Jason Weston, Léon Bottou, Michael Karlen, Koray Kavukcuoglu, and Pavel Kuksa. Natural language processing (almost) from scratch. *J. Mach. Learn. Res.*, 12:2493–2537, 2011.
10. Zhiheng Huang, Wei Xu, and Kai Yu. Bidirectional LSTM-CRF models for sequence tagging. *CoRR*, abs/1508.01991, 2015.
11. Guillaume Lample, Miguel Ballesteros, Sandeep Subramanian, Kazuya Kawakami, and Chris Dyer. Neural architectures for named entity recognition. In *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 260–270, 2016.
12. Xuezhe Ma and Eduard Hovy. End-to-end sequence labeling via bi-directional lstm-cnns-crf. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1064–1074, 2016.
13. Yijia Liu, Yue Zhang, Wanxiang Che, Ting Liu, and Fan Wu. Domain adaptation for crf-based chinese word segmentation using free annotations. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 864–874, 2014.
14. Jingbo Shang, Liyuan Liu, Xiang Ren, Xiaotao Gu, Teng Ren, and Jiawei Han. Learning named entity tagger using domain-specific dictionary. In *EMNLP*, 2018.
15. David D Lewis and Jason Catlett. Heterogeneous uncertainty sampling for supervised learning. In *Machine learning proceedings 1994*, pages 148–156. Elsevier, 1994.
16. H. S. Seung, M. Opper, and H. Sompolinsky. Query by committee. In *Proceedings of the Fifth Annual Workshop on Computational Learning Theory*, pages 287–294, 1992.
17. Andrew McCallum and Kamal Nigam. Employing em and pool-based active learning for text classification. In *Proceedings of the Fifteenth International Conference on Machine Learning*, pages 350–358, 1998.
18. Burr Settles, Mark Craven, and Soumya Ray. Multiple-instance active learning. In *Advances in neural information processing systems*, pages 1289–1296, 2008.
19. Nicholas Roy and Andrew McCallum. Toward optimal active learning through sampling estimation of error reduction. In *ICML*, 2001.
20. Barun Patra and Joel Ruben Antony Moniz. Weakly supervised attention networks for entity recognition. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP- IJCNLP)*, pages 6268–6273, 2019.
21. An Thanh Nguyen, Byron Wallace, Junyi Jessy Li, Ani Nenkova, and Matthew Lease. Aggregating and predicting sequence labels from crowd annotations. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 299–309, 2017.
22. Pierre Lison, Jeremy Barnes, Aliaksandr Hubin, and Samia Touileb. Named entity recognition without labelled data: A weak supervision approach. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 1518–1533. Association for Computational Linguistics, 2020.
23. Graham Neubig, Yosuke Nakata, and Shinsuke Mori. Pointwise prediction for robust, adaptable japanese morphological analysis. In *The 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies (ACL-HLT)*, pages 529–533, 2011.
24. Tobias Scheffer, Christian Decomain, and Stefan Wrobel. Active hidden markov models for in- formation extraction. In *International Symposium on Intelligent Data Analysis*, pages 309–318, 2001.