Clustering-based Inference for Biomedical Entity Linking

Rico Angell[1], Nicholas Monath[1], Sunil Mohan[2], Nishant Yadav[1], and Andrew McCallum[1]
[1] College of Information and Computer Sciences
University of Massachusetts Amherst
{rangell,nmonath,nishantyadav,mccallum}@cs.umass.edu
[2]Chan Zuckerberg Initiative
smohan@chanzuckerberg.com

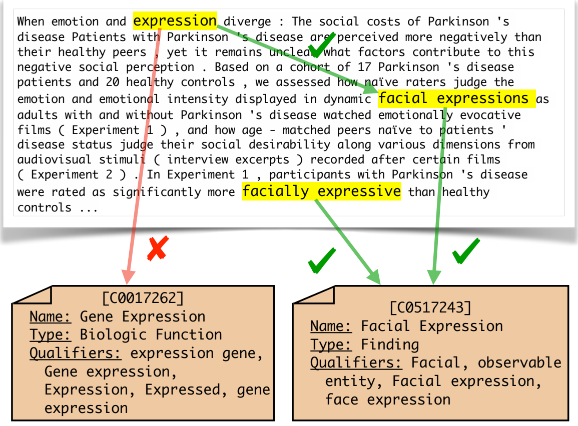
Abstract

Due to large number of entities in biomedical knowledge bases, only a small fraction of entities have corresponding labelled training data. This necessitates entity linking models which are able to link mentions of unseen entities using learned representations of entities. Previous approaches link each mention independently, ignoring the relationships within and across documents between the entity mentions. These relations can be very useful for linking mentions in biomedical text where linking decisions are often difficult due mentions having a generic or a highly specialized form. In this paper, we introduce a model in which linking decisions can be made not merely by linking to a knowledge base entity but also by grouping multiple mentions together via clustering and jointly making linking predictions. In experiments we improve the state-of-the-art entity linking accuracy on two biomedical entity linking datasets including on the largest publicly available dataset.

1 Introduction

Ambiguity is inherent in the way entities are mentioned in natural language text. Grounding such ambiguous mentions to their corresponding entities, the task of *entity linking*, is critical to many applications: automated knowledge base construction and completion (Riedel et al., 2013; Surdeanu et al., 2012), information retrieval (Meij et al., 2014), smart assistants (Balog and Kenter, 2019), question answering (Dhingra et al., 2020), text mining (Leaman and Lu, 2016; Murty et al., 2018).

Consider the excerpt of text from a biomedical research paper in Figure 1, the three highlighted mentions (*expression*, *facial expressions*, and *facially expressive*) all link to the same entity, namely `C0517243 - Facial Expresson` in the leading biomedical KB, Unified Medical Language System (UMLS) (Bodenreider, 2004).



When emotion and expression diverge : The social costs of Parkinson 's disease Patients with Parkinson 's disease are perceived more negatively than their healthy peers , yet it remains unclear what factors contribute to this negative social perception . Based on a cohort of 17 Parkinson 's disease patients and 20 healthy controls , we assessed how naïve raters judge the emotion and emotional intensity displayed in dynamic facial expressions as adults with and without Parkinson 's disease watched emotionally evocative films (Experiment 1) , and how age - matched peers naïve to patients ' disease status judge their social desirability along various dimensions from audiovisual stimuli (interview excerpts) recorded after certain films (Experiment 2) . In Experiment 1 , participants with Parkinson 's disease were rated as significantly more facially expressive than healthy controls ...

[C0017262]
Name: Gene Expression
Type: Biologic Function
Qualifiers: expression gene, Gene expression, Expression, Expressed, gene expression

[C0517243]
Name: Facial Expression
Type: Finding
Qualifiers: Facial, observable entity, Facial expression, face expression

Figure 1: **Biomedical Entity Linking**. All three highlighted mentions refer to the same entity. The mention `expression` is clearly related to the other two highlighted mentions which are much less ambiguous. If considered independently `expression` is more closely related to an incorrect entity.

The mention *expression* is highly ambiguous and easily confused with the more prevalent entity, `Gene expression`. This linking decision may become easier with sufficient training examples (or sufficiently rich structured information in the knowledge-base) . However, in biomedical (Mohan and Li, 2019) and other specialized domains (Logeswaran et al., 2019), it is often the case that the knowledge-base information is largely incomplete. Furthermore, the scarcity of training data leads to a setting in which most entities have not been observed at training.

State-of-the-art entity linking methods which are able to link entities unseen at training time make predictions for each mention independently (Logeswaran et al., 2019; Wu et al., 2019). In this way, the methods may have difficulty linking mentions which, as in the example above, have little lexical similarity with the entities in the knowledge-base, as well as mentions for which the context is highly ambiguous. These mentions cannot directly use information from one mention (or its linking decision) to inform the prediction of another mention. On the other hand, entity linking methods that do

Proceedings of the 2021 Conference of the North American Chapter of the
Association for Computational Linguistics: Human Language Technologies, pages 2598–2608
June 6–11, 2021. ©2021 Association for Computational Linguistics

jointly consider entity linking decisions (Ganea and Hofmann, 2017; Le and Titov, 2018) are designed for cases in which all of the entities in the knowledge-base to have example mentions or metadata at training time (Logeswaran et al., 2019).

In this paper, we propose an entity linking model in which entity mentions are either (1) linked directly to an entity in the knowledge-base or (2) join a cluster of other mentions and link as a cluster to an entity in the knowledge-base. Some mentions may be difficult to link directly to their referent ground truth entity, but may have very clear coreference relationships to other mentions. So long as one mention among the group of mentions clustered together links to the correct entity the entire cluster can be correctly classified. This provides for a joint, tranductive-like, inference procedure for linking. We describe both the inference procedure as well as training objective for optimizing the model's inference procedure, based on recent work on supervised clustering (Yadav et al., 2019).

It is important to note that our approach does not aim to do joint coreference and linking, but rather makes joint linking predictions by clustering together mentions that are difficult to link directly to the knowledge-base. For instance, in Figure 1, the mention *expression* may be difficult to link to the ground truth `Facial expression` entity in the knowledge-base because the mention can refer to a large number of entities. However, the local syntactic and semantic information of the paragraph give strong signals that *expression* is coreferent with *facial expression*, which is easily linked to the correct entity.

We perform experiments on two biomedical entity linking datsets: MedMentions (Mohan and Li, 2019), the largest publicly available dataset as well as the benchmark BC5CDR (Li et al., 2016). We find that our approach improves over our strongest baseline by 2.3 points of accuracy on MedMentions and 0.8 points of accuracy on BC5CDR over the baseline method (Logeswaran et al., 2019). We further analyze the performance of our approach and observe that (1) our method better handles ambiguous mention surface forms (as in the example shown in Figure 1) and (2) our method can correctly link mentions even when the candidate generation step fails to provide the correct entity as a candidate.

2 Background

Each document $D \in \mathcal{D}$, has a set of mentions $\mathcal{M}^{(D)} = \{m_1^{(D)}, m_2^{(D)}, \ldots, m_N^{(D)}\}$. We denote the set of all mentions across all documents as plainly $\mathcal{M}$. The task of entity linking is to classify each mention m_i as referent to a single entity e_i from a KB of entities. We use $\mathcal{E}(m_i)$ to refer to the ground truth entity of mention m_i and $\hat{e}_i$ to refer to the predicted entity.

Knowledge-bases. We assume that we are given a knowledge-base corresponding to a closed world of entities. These KBs are typically massive: English Wikipedia contains just over 6M entities[1] and the 2020 release of the UMLS contains 4.28M entities[2]. We describe in Sections 5.1 & 5.2 the details of the KBs used in each of the experiments.

Candidate Generation. Given the massive number of entities to which a mention may refer, previous work (Logeswaran et al., 2019, inter alia) uses a candidate generation step to reduce the restrict the number of entities considered for a given mention, m, to a candidate set $\Gamma(m)$. The recall of this step is critical to the overall performance of entity linking models.

3 Model

In this section, we describe our clustering-based approach for jointly making entity linking predictions for a set of mentions. Our proposed inference method builds a graph where the nodes are the union of all of the mentions and entities and the edges have weights denoting the affinities between the endpoints. To make linking decisions, we cluster the nodes of the graph such that each cluster contains exactly one entity, following which each mention is assigned to the entity in its cluster.

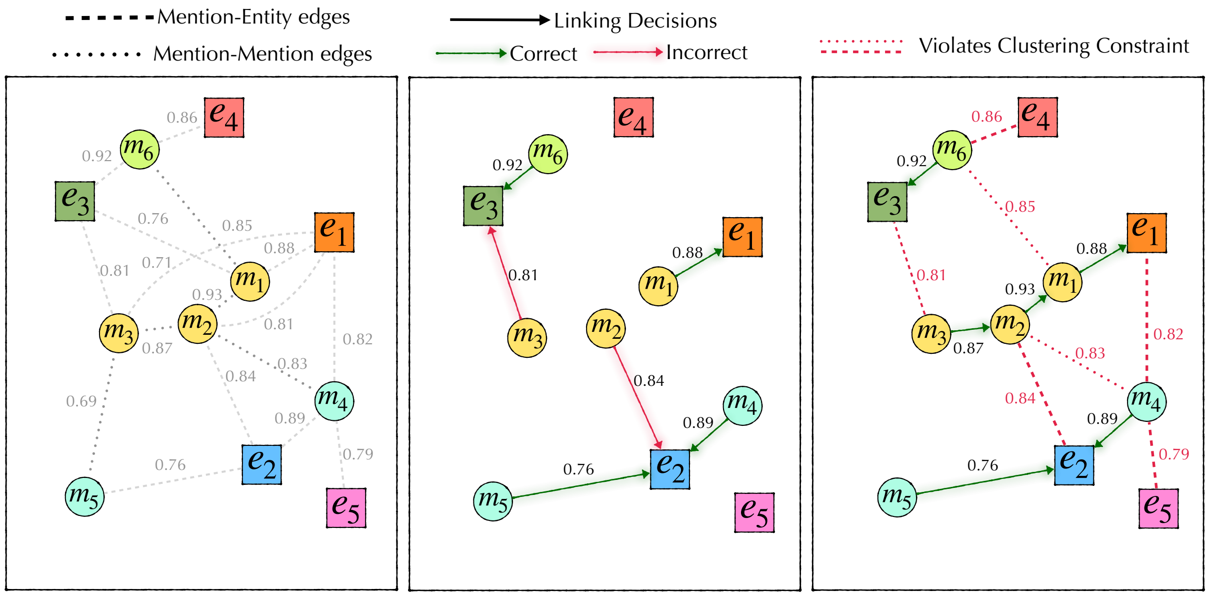
3.1 Clustering-based Entity Linking

Let $\varphi : \mathcal{M} \times \mathcal{M} \to \mathbb{R}$ and $\psi : \mathcal{M} \times \mathcal{E} \to \mathbb{R}$ be parameterized functions which compute symmetric mention-mention and mention-entity affinities, respectively. The exact parameterizations of these functions are detailed in Section 3.2.

Define the undirected, weighted graph $G = (V, E, w)$ where $V = \mathcal{M} \cup \mathcal{E}$ and $E = \mathcal{M} \times$

[1] number of content pages as of May 20, 2020, `https://en.wikipedia.org/wiki/Special:Statistics`

[2] `https://www.nlm.nih.gov/research/umls/knowledge_sources/metathesaurus/release/notes.html`



Graph w/ mention-mention and mention-entity affinities

Independently linking mentions to entities

Linking mention to entities with our proposed inference

Figure 2: **Clustering-based Inference for Entity Linking**. Mentions are shown in circles and entities in squares. Color families indicate ground-truth cluster assignments. The left figure shows the graph G that is the basis of the clustering task, the center figure show predictions under independent linking model, and the right figure shows our proposed inference linking mentions to entities by running our proposed constrained clustering inference procedure over G that assigns at most one entity per cluster.

$\mathcal{M} \cup \{(m, e) : e \in \Gamma(m)\}$. The weight of each edge, $w(v_i, v_j)$ for $v_i, v_j \in V$, is determined by φ or ψ depending on the vertices of the edge: $w(m_i, m_j) = \varphi(m_i, m_j)$ and $w(m_i, e_l) = \psi(m_i, e_l)$. Linking decisions for each mention are determined by clustering the vertices of G under the constraint that every entity must appear in exactly one cluster.

Given the graph G, we start with every node in their own individual cluster. We define affinity between a pair of clusters as the strongest cross-cluster edge between nodes in the two clusters. Iteratively, we greedily merge clusters by choosing a pair of clusters with largest affinity between them under the constraint that we cannot merge two clusters which *both* contain an entity. When every cluster contains exactly one entity, this process can no longer merge any clusters, and thus terminates[3]. Each mention is linked to the entity present in its cluster at the end of inference. Algorithm 1 describes this process of constructing the graph and clustering nodes to make linking decisions more formally.

Figure 2 shows the proposed inference in action on five entities and six mentions. Initially, every mention and entity start in a singleton cluster. In the first round, clusters $\{m_1\}$ and $\{m_2\}$ are merged, followed by merger of $\{e_3\}$ and $\{m_6\}$ in the second round, and so on. Note that in fifth round, clusters $c_1 = \{m_4, e_2\}$ has higher affinity with $c_2 = \{m_1, m_2, m_3, e_1\}$ than with $c_3 = \{m_5\}$, yet c_1 and c_3 are merged instead of c_1 and c_2 due to the constraint that we cannot merge two clusters which *both* contain an entity. At the end, every mention is clustered together with exactly one entity, and there could be entities present as singleton clusters such as $\{e_4\}$ and $\{e_5\}$. Note that m_3 correctly links to its gold entity e_1 as a result of being clustered with mentions m_1, m_2 even though it has higher affinity with entity e_3 : $w(m_3, e_3) > w(m_3, e_1)$.

3.2 Affinity Models

We parameterize $\psi(\cdot, \cdot)$ and $\phi(\cdot, \cdot)$ using two separate deep transformer encoders (Vaswani et al., 2017) for our mention-mention affinity model and mention-entity affinity model — specifically we use the BERT architecture (Devlin et al., 2019) initialized using the weights from BioBERT (Lee et al., 2019).

[3] This process is equivalent to single-linkage hierarchical agglomerative clustering with the constraint that two entities cannot be in the same cluster.

3.2.1 Mention-Mention Model

The mention-mention model is also a cross-encoder, taking as input a pair of mention in context and producing a single scalar affinity for every pair. The input tokens take the form:

$$\texttt{[CLS]} < m_i > \texttt{[SEP]} < m_j > \texttt{[SEP]}$$
$$\text{where } < m_i > := c_l \, \texttt{[START]} \, m_i \, \texttt{[END]} \, c_r$$

where m_i is the mention tokens and c_l and c_r are the left and right context of the mention in the text, respectively. The `[START]` and `[END]` tokens are special tokens fine-tuned to signify the start and end of the mention in context, respectively. We restrict the length of each input sequence to have a maximum of 256 tokens. A representations of each mention is computed using the average of the encoder's output representations corresponding to the mention's input tokens. The affinity for a mention pair is computed by concatenating their mention representations and passing it through a linear layer with a sigmoid activation. We make this affinity symmetric by averaging the two possible orderings of a pair of mentions in the cross-encoder input sequence.

3.2.2 Mention-Entity Model

The mention-entity affinity model is a cross-encoder model (Vig and Ramea, 2019; Wolf et al., 2019; Humeau et al., 2019, inter alia) and takes as input the concatenation of the mention in context with the entity description. The input tokens take the form:

$$\texttt{[CLS]} \, c_l \, \texttt{[START]} \, m \, \texttt{[END]} \, c_r \, \texttt{[SEP]} \, e \, \texttt{[SEP]}$$

where the mention in context is the same as in the mention-mention model and e is the description of the entity. We restrict the length of this input sequence to 256 tokens. After passing the input sequence through BERT, we transform the output representation corresponding to the `[CLS]` token with a linear layer with one output unit. This value is finally passed through the sigmoid function to output affinity between the mention and the entity.

4 Training

In this section, we explain the training procedure for the affinity models $\varphi(\cdot,\cdot)$ and $\psi(\cdot,\cdot)$ used by the clustering inference procedure. We train the mention-mention and mention-entity models independently in a way that allows the affinities to be comparable when performing inference.

Algorithm 1 Clustering Inference for Linking

```
Input: $(\mathcal{M}, \mathcal{E}, \Gamma, \varphi, \psi)$
Output: $\{(m_i, \hat{e}_i)\}_{i=1}^{|\mathcal{M}|}$
▷ Construct the graph $G$
$E = \{\}$
for $m_i \in \mathcal{M}$ do
    Let $D_i$ be the document containing $m_i$
    for $m_j \in \mathcal{M}^{(D_i)} \setminus \{m_i\}$ do
        $E = E \cup \{(m_i, m_j, \varphi(m_i, m_j))\}$
    for $e_l \in \Gamma(m_i)$ do
        $E = E \cup \{(m_i, e_l, \psi(m_i, e_l))\}$
Construct $G = (V, E)$ from edge set $E$
Let $S$ be the edges sorted in descending order
▷ Cluster nodes of $G$ under linking constraint
$\hat{\mathcal{C}} = \{\{v\} | v \in V\}$
for $(s, t) \in S$ do
    if $\hat{\mathcal{C}}(s) \cap \mathcal{E} = \emptyset$ or $\hat{\mathcal{C}}(t) \cap \mathcal{E} = \emptyset$ then
        $\hat{\mathcal{C}} = \hat{\mathcal{C}} \setminus \{\hat{\mathcal{C}}(s), \hat{\mathcal{C}}(t)\}$
        $\hat{\mathcal{C}} = \hat{\mathcal{C}} \cup \{\hat{\mathcal{C}}(s) \cup \hat{\mathcal{C}}(t)\}$
▷ Make linking decisions based on clustering
$L = \{\}$
for $C \in \hat{\mathcal{C}}$ do
    $M = C \cap \mathcal{M}$
    $\{\hat{e}\} = C \cap \mathcal{E}$
    for $m \in M$ do
        $L = L \cup \{(m, \hat{e})\}$
return $L$
```

We use triplet max-margin based training objectives to train both models. The most important aspect of our procedure is how we pick negatives during training. For the mention-entity model, we restrict our negatives to be from the candidate set. For the mention-mention model, we restrict our negatives to come from mentions within the same document. From these sets of possible negatives we choose the top-k most offending ones according the instantaneous state of the model – i.e. the negatives with highest predicted affinities according to the model at that point during training. The following sections detail the training procedures for both models.

4.1 Mention-Mention Affinity Training

To train the mention-mention affinity model we use a variant of the maximum spanning tree (MST) supervised single linkage clustering algorithm presented in Yadav et al. (2019). Let $\mathcal{M}_{e_l}^{(D)} = \{m \in \mathcal{M}^{(D)} \,|\, \mathcal{E}(m) = e_l\}$ be the set of mentions referring to entity e_l in any one document and the set of ground truth clusters be represented by

$\mathcal{C}^* = \{\mathcal{M}_{e_l}^{(D)} \,|\, e_l \in \mathcal{E}\}$. Let P be the set of positive training edges: the edges of the MST of the complete graph on the cluster $C \in \mathcal{C}^*$. Let $N_\varphi(m_*)$ be the k-nearest within document negatives to the anchor point $m_* \in C$ according to the current state of the model during training. The objective of this training procedure is to minimize the following triplet max-margin loss[4] with margin μ for each cluster $C \in \mathcal{C}^*$:

$$\mathcal{L}_\varphi(\theta; C) = \sum_{m_*, m_+ \in P} \; \sum_{m_- \in N_\varphi(m_*)} \ell_{\varphi,\mu}(m_*, m_+, m_-),$$

where $\ell_{\varphi,\mu}(a, p, n) = [\varphi(a, n) - \varphi(a, p) + \mu]_+$.

4.2 Mention-Entity Affinity Training

For the mention-entity model, we use a triplet max-margin based objective with margin μ where the anchor is a mention m in the training set, the positive is the ground truth entity $e_+ = \mathcal{E}(m)$, and the negatives are chosen from the candidate set $\Gamma(m)$. Denote the k most offending negatives according to the current state of the model during training as $N_\psi(m) \subseteq \Gamma(m) \setminus \{\mathcal{E}(m)\}$. Formally, the loss is

$$\mathcal{L}_\psi(\theta; \mathcal{M}) = \sum_{m, e_+} \; \sum_{e_- \in N_\psi(m)} \ell_{\psi,\mu}(m, e_+, e_-),$$

where $\ell_{\psi,\mu}(a, p, n) = [\psi(a, n) - \psi(a, p) + \mu]_+$.

5 Experiments

We evaluate on biomedical entity linking using the MedMentions (Mohan and Li, 2019) and BC5CDR (Li et al., 2016) datasets. We compare to state-of-the-art methods. We then analyze the performance of our method in more detail and provide qualitative examples demonstrating our approaches' ability to use mention-mention relationships to improve candidate generation and linking.

5.1 MedMentions

MedMentions is a publicly available[5] dataset consisting of the titles and abstracts of 4,392 PubMed articles. The dataset is hand-labeled by annotators and contains labeled mention spans and entities linked to the 2017AA full version of UMLS. Following the suggestion of Mohan and Li (2019), we use the ST21PV subset, which restricts the entities linked in documents to a set of 21 entity types that were deemed most important for building scientific knowledge-bases. We refer the readers to Mohan and Li (2019) for a complete analysis of the dataset and provide a few important summary statistics here. The train/dev/test split partitions the PubMed articles into three non-overlapping groups. This means that some entities seen at training time will appear in dev/test and other entities will appear in dev/test but not at training time. In fact, a large number of entities that appear in dev/test time are unseen at training, about 42% of entities. See Table 1 for split details and statistics.

	MedMentions			**BC5CDR**		
	Train	**Dev**	**Test**	**Train**	**Dev**	**Test**
$\lvert\mathcal{M}\rvert$	120K	40K	40K	18K	934	10K
$\lvert\mathcal{E}(\mathcal{M})\rvert$	19K	9K	8K	2K	281	1K
% seen	100	57.7	57.5	100	80.1	64.8

Table 1: **Linking Datasets**. Statistics of each dataset, including the percent of ground truth entities seen during training (% seen).

Previous work has evaluated on MedMentions using unfairly optimistic candidate generation settings such as using only 10 candidates including the ground truth (Zhu et al., 2019) or restricting candidates to entities appearing somewhere in the MedMentions corpus (Murty et al., 2018). We instead work in a much more general setting where all entities in UMLS are considered at candidate generation time and the generated candidates might not include the ground truth entity.

5.2 BC5CDR

BC5CDR (Li et al., 2016) is another entity linking benchmark in the biomedical domain. The dataset consists of 1,500 PubMed articles annotated with labeled disease and chemical entities. Unlike MedMentions, which contains 21 types of entities, this dataset contains just two types. These chemical and disease mentions are labeled with entities from MeSH[6], a much smaller biomedical KB than UMLS. See Table 1 for split details and statistics.

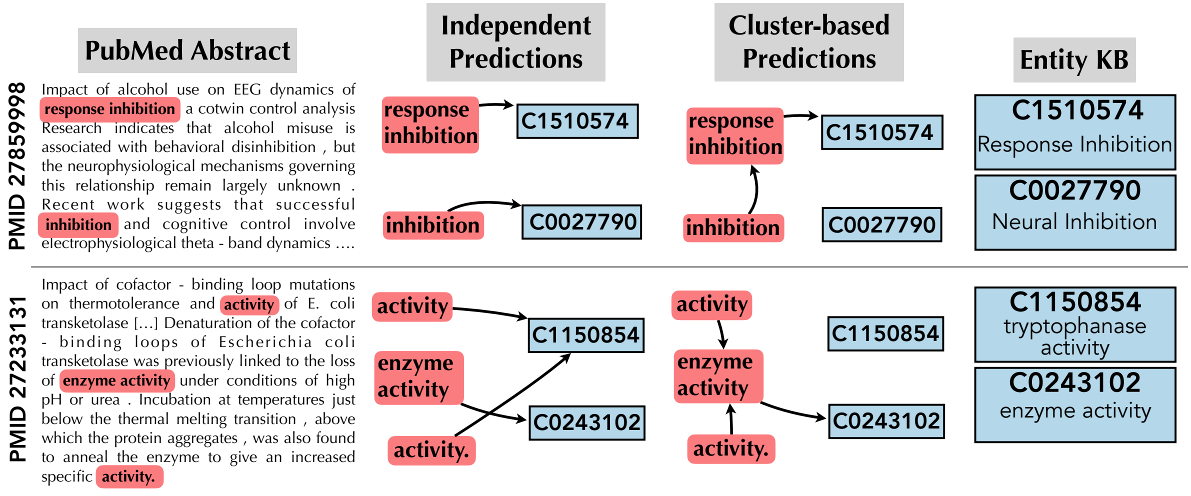
5.3 Preprocessing

The MedMentions ST21PV corpus is processed as follows: (i) Abbreviations defined in the text of each paper are identified using AB3P (Sohn et al., 2008). Each definition and abbreviation instance is then replaced with the expanded form. (ii) The text of each paper in the corpus is tokenized and

[4] Define $[x]_+ = \max(x, 0)$

[5] https://github.com/chanzuckerberg/MedMentions

[6] https://www.nlm.nih.gov/mesh



Impact of alcohol use on EEG dynamics of **response inhibition** a cotwin control analysis Research indicates that alcohol misuse is associated with behavioral disinhibition , but the neurophysiological mechanisms governing this relationship remain largely unknown . Recent work suggests that successful **inhibition** and cognitive control involve electrophysiological theta - band dynamics ....

Impact of cofactor - binding loop mutations on thermotolerance and **activity** of E. coli transketolase [...] Denaturation of the cofactor - binding loops of Escherichia coli transketolase was previously linked to the loss of **enzyme activity** under conditions of high pH or urea . Incubation at temperatures just below the thermal melting transition , above which the protein aggregates , was also found to anneal the enzyme to give an increased specific **activity.**

Figure 3: **Example predictions on Ambiguous Mentions**. Here we show two example outputs for highly ambigous mention surface forms (`inhibition` and `activity`). The independent model incorrectly makes predictions on these surface forms. The clustering-based model is able to have each ambiguous mention link to a less ambiguous mention in the same abstract and thereby make correct predictions.

split into sentences using CoreNLP (Manning et al., 2014). (iii) Overlapping mentions are resolved by preferring longer mentions that begin earlier in each sentence, and mentions are truncated at sentence boundaries. This results in 379 mentions to be dropped from the total of 203,282. (iv) Finally, the corpus is saved into the IOB2 tag format. The same preprocessing steps are used for BC5CDR, except overlapping mentions are not dropped.

5.4 Candidate Generation

For both datasets, we use a character n-gram TF-IDF model to produce candidates for all of the mentions in all splits. The candidate generator utilizes the 200k most frequent character n-grams, $n \in \{2 \ldots 5\}$ and the 200k most frequent words in the names in $\mathcal{E}$ to produce sparse vectors for all of the mentions and entity descriptions (which in our case is the canonical name, the type, and a list of known aliases and synonyms). Table 5 provides candidate generation results for each dataset. The results report the average recall@K at different numbers of candidates (K), i.e., whether or not the gold entity is top K candidates for a given mention.

5.5 Training and Inference Details

Our model contains 220M parameters, the majority of which are contained within the two separate BERT-based models. We optimize both the models with mini-batch stochastic gradient descent using the Adam optimizer (Kingma and Ba, 2014) with recommended learning rate of 5e-5 (Devlin et al., 2019) with no warm-up steps. We accumulate gradients over all of the triples for a batch size of 16 within document clusters. We compute the top-k most offending negatives on-the-fly for each batch by running the model in inference mode proceeding each training step. Training and inference are done on a single machine with 8 NVIDIA 1080 Ti GPUs. We train our model on MedMentions for two epochs and BC5CDR for four epochs. Training takes approximately three days for MedMentions and one day for BC5CDR. Clustering-based inference takes about three hours for MedMentions and one hour for BC5CDR. Code and data to reproduce experiments will be made available.

5.6 Results

We compare our clustering-based inference procedure, which we refer to our approach as CLUSTERING-BASED, to a state-of-the-art independent inference procedure, INDEPENDENT, which is the zero-shot architecture proposed by Logeswaran et al.. This same model is used as the mention-entity affinity model used in our approach. We also compare to to an n-gram tf-idf model (our candidate generation model), TAGGERONE (Leaman and Lu, 2016), BIOSYN (Sung et al., 2020), and SAPBERT (Liu et al., 2020) on both MedMentions and BC5CDR.

Table 2 shows performance of the baseline models, INDEPENDENT, and CLUSTERING-BASED inference procedure on MedMentions and BC5CDR. We report results using the gold mention segmen-

	MedMentions			BC5CDR		
	Overall	Acc. on		Overall	Acc. on	
	Acc.	Seen	Unseen	Acc.	Seen	Unseen
N-gram TF-IDF	50.9	50.9	51.0	86.9	89.2	74.6
BioSyn (Sung et al., 2020)	72.5†	76.5†	58.7†	87.8†	89.0†	81.1†
SapBERT (Liu et al., 2020)	69.8†	72.9†	58.9†	85.2†	85.8†	**82.0**†
Independent (Logeswaran et al., 2019)	72.8	75.9	61.9	90.5	94.0	73.6
Clustering-based (ours)	**74.1**	**77.3**	**62.9**	**91.3**	**94.9**	73.8
w/ Gold Types						
N-gram TF-IDF	67.9	69.0	64.0	87.8	90.2	76.1
TaggerOne (Leaman and Lu, 2016)	73.8	78.2	58.8	89.8	91.8	79.9
BioSyn (Sung et al., 2020)	77.0†	80.7†	64.1†	87.9†	89.1†	81.3†
SapBERT (Liu et al., 2020)	74.1†	77.0†	63.8†	86.0†	86.8†	**82.0**†
Independent (Logeswaran et al., 2019)	76.8	79.2	68.4	90.6	94.1	73.6
Clustering-based (ours)	**79.1**	**81.5**	**70.5**	**91.4**	**94.9**	74.0

Table 2: **Entity Linking Results**. We report linking accuracy on MedMentions and BC5CDR datasets with gold mentions spans and gold mention spans and types. We observe that Clustering-based inference provides improved accuracy in each setting with additional improvements seen when gold entity types are provided. (†Hits at one synonym — multiple entities could be predicted)

tation (rather than end-to-end) to focus on the performance of each model in terms of linking rather than confounding the performance by including segmentation. Due to TaggerOne's joint entity recognition, typing, and linking architecture, we cannot make predictions for gold mention boundaries without also using their gold types. And so to have a fair comparison to TaggerOne, we provide the gold mention boundaries and types to each system and report these results as well.

We use *seen* and *unseen* to refer to the sets of mentions whose ground truth entities are seen and unseen at training, respectively. Note that even if a mention is in the subset of mentions referred to as *seen*, it does not mean that we have seen the particular surface form before in the training set, merely that we have seen other mentions of that particular entity.

On MedMentions, when the models are provided with only the gold mention span, Clustering-based inference procedure outperforms Independent by 1.3 points of accuracy, and we see improvements in accuracy for both seen and unseen entities. When the models are additionally provided with the gold type, we see substantial improvements in accuracy for both Independent and Clustering-based over TaggerOne, namely 3.0 and 5.3 points of improvement, respectively.

On BC5CDR, when the models are provided with only the gold mention span, Clustering-based inference procedure outperforms Independent by 0.4 points of accuracy, and we see improvements in accuracy for both seen and unseen entities. When the models are additionally provided with the gold type, we see improvements in accuracy for both Independent and Clustering-based over TaggerOne, namely 0.8 and 1.6 points of improvement, respectively.

Observe that the candidate generation results are drastically different for the two datasets (Table 5). We posit that the ability to generate correct candidates correlates with the relative difficultly of the linking task on each dataset, respectively.

5.7 Analysis: Recovering from Poor Candidate Generation

We hypothesize that our clustering-based inference procedure would allow for better performance on mentions for which candidate generation is difficult. Observe that while the performance of the independent model is upper bounded by the recall of candidate generation, this is not an upper bound for the clustering-based model. The clustering-based model can allow mentions that have no suitable candidates to link to other mentions in the same document. We report the accuracy of both systems with respect to whether or not the ground truth entity is in each mentions' list of candidates.

The accuracy for each system and each partition of mentions is shown in Table 3. Observe that our approach offers a large number of mentions a correct resolution, when the independent model could not link them correctly due to the ground truth entity being missing from the candidate list. Additionally, it can be seen that CLUSTERING-BASED does sacrifice some performance in comparison to INDEPENDENT, but more than makes up for it in the case where the ground truth entity is not in the candidate set.

5.8 Analysis: Handling Ambiguous Mentions

We also hypothesize that for mentions which are highly ambiguous and could refer to many different entities, such as common nouns like *virus*, *disease*, etc, the clustering-based inference should offer improvements. Table 4 shows that our approach is able to correctly link more ambiguous mentions compared to independent model[7]. Figure 3 shows two examples from this subset where CLUSTERING-BASED inference is able to make the correct linking decision and INDEPENDENT is not.

6 Related Work

Entity linking is widely studied and often focused on linking mentions to Wikipedia entities (also known as Wikification) (Mihalcea and Csomai, 2007; Cucerzan, 2007; Milne and Witten, 2008; Hoffart et al., 2011; Ratinov et al., 2011; Cheng and Roth, 2013). Entity linking is often done independently for each mention in the document (Ratinov et al., 2011; Raiman and Raiman, 2018) or by modeling dependencies between predictions of entities in a document (Cheng and Roth, 2013; Ganea and Hofmann, 2017; Le and Titov, 2018).

In the biomedical domain, Unified Medical Language System (UMLS) is often used as a knowledge-base for entities (Mohan and Li, 2019; Leaman and Lu, 2016). While UMLS is a rich ontology of concepts and relationships between them, this domain is low resource compared to Wikipedia with respect to number of labeled training data for each entity mention. This leads to a zero-shot setting in datasets such as MedMentions (Mohan and Li, 2019) where new entities are seen at test time. Previous work has addressed this zero-shot setting using models of the type hierarchy (Murty et al., 2018; Zhu et al., 2019). This previous work (Murty et al., 2018; Zhu et al., 2019) uses an unrealistic candidate generation setting where the true positive candidate is within the candidate set and/or entities are limited to those in the dataset rather than those in the knowledge-base.

Mention-mention relationships are also explored in (Le and Titov, 2018) which extends the pairwise CRF model (Ganea and Hofmann, 2017) to use mention-level relationships in addition to entity relationships. These works use attention in a way to build the context representation of the mentions. However, as mentioned by Logeswaran et al. (2019) is not well suited for zero-shot linking.

Coreference (both within and across documents) has also been explored by past work (Dutta and Weikum, 2015). This work uses an iterative procedure that performs hard clustering for the sake of aggregating the contexts of entity mentions. Durrett and Klein (2014) presents a CRF-based model for joint NER, within-document coreference, and linking. They show that jointly modeling these three tasks improves performance over the independent baselines. This differs from our work since we do not require coreference decisions to be correct in order to make correct linking decisions. Other work performs joint entity and event coreference (Barhom et al., 2019) without linking.

7 Conclusion

In this work, we presented a novel clustering-based inference procedure which enables joint entity linking predictions. We evaluate the effectiveness of our approach on the two biomedical entity linking datasets, including the largest publicly available dataset. We show through analysis that our approach is better suited to link mentions with ambiguous surface forms and link mentions where the ground truth entity is not in the candidate set.

8 Ethical Considerations

Entity linking is a task with the intention of providing useful information when building a semantic index of documents. This semantic index is a core component of systems which allow users to search, retrieve, and analyze text documents. In our specific case, we are interested in building semantic indexes of scientific documents where the end user would be scientists and researchers. The goal is to help them navigate the vast amount of literature and accelerate science. This being said, users need to take the outputs of such a system as suggestions and with the potential that the information is incorrect. Researchers must be aware that the sys-

[7] These are: *activation, activity, a, b, cardiac, cells, clinical, compounds, cr, development, disease, function, fusion, inhibition, injuries, injury, liver, management, methods, mice, model, pa, production, protein, regulation, report, responses, response, r, screening, stress, studies, study, treatment*

	MedMentions		BC5CDR	
	$\mathcal{E}(m) \in \Gamma(m)$	$\mathcal{E}(m) \notin \Gamma(m)$	$\mathcal{E}(m) \in \Gamma(m)$	$\mathcal{E}(m) \notin \Gamma(m)$
INDEPENDENT	**85.3**	0.0	**95.5**	0.0
CLUSTERING-BASED	84.5	**13.9**	95.3	**14.9**
w/ Gold Types				
INDEPENDENT	**90.0**	0.0	**95.7**	0.0
CLUSTERING-BASED	89.3	**19.3**	95.4	**15.9**

Table 3: **Performance when Candidate Generation Fails**. We report the accuracy of each method on mentions for which the ground truth entity is in the candidate list ($\mathcal{E}(m) \in \Gamma(m)$) and is not in the list ($\mathcal{E}(m) \notin \Gamma(m)$). We observe that our proposed approach is able to perform reasonably well even when candidate generation fails.

	Accuracy
INDEPENDENT (Logeswaran et al., 2019)	71.91
CLUSTERING-BASED (ours)	**73.03**

Table 4: **Performance on Ambiguous Mentions** We select mentions for which the surface form is labeled 10 or more different entities in MedMentions and measure performance on instances of these surface forms on the test data. We observe that CLUSTERING-BASED is able to more accurately link these mentions. Figure 3 shows examples of these mentions.

Recall@	BC5CDR	MedMentions
1	86.9	50.8
2	89.4	63.8
4	91.1	73.4
8	92.1	79.2
16	93.1	82.3
32	94.3	84.6
64	94.9	85.3

Table 5: **Candidate Generation Recall**. Recall is measured by whether or not the ground truth entity is in the top K candidate entities for the given mention. We report the micro average recall over all mentions.

tem is not perfect and they should not jump to any conclusions especially about important decisions. Additionally, the researcher can always verify the decisions being made by the system.

While this paper focuses on biomedical entity linking, this technique could be extended to other domains. In such other domains, users might not have as much expertise, but the user is still responsible for making decisions on their own, since the system is not perfect. In addition, the system developers and designers need to be aware of their particular application to ensure to mitigate harm which could come from such a system. For example, in any application that deals with personalized data, we need to be wary of the potential outcomes which could come from an entity linking based system or semantic index, such as privacy or other potential malicious behaviour or unforeseen consequences due to the decisions being made by the system.

Acknowledgements

We thank members of UMass IESL and NLP groups for helpful discussion and feedback. This work is funded in part by the Center for Data Science and the Center for Intelligent Information Retrieval, and in part by the National Science Foundation under Grants No. 1763618, and in part by the Chan Zuckerberg Initiative under the project Scientific Knowledge Base Construction. The work reported here was performed in part by the Center for Data Science and the Center for Intelligent Information Retrieval, and in part using high performance computing equipment obtained under a grant from the Collaborative R&D Fund managed by the Massachusetts Technology Collaborative. Rico Angell is supported by the National Science Foundation Graduate Research Fellowship under Grant No. 1938059. Any opinions, findings and conclusions or recommendations expressed in this material are those of the authors and do not necessarily reflect those of the sponsor.

References

Krisztian Balog and Tom Kenter. 2019. Personal knowledge graphs: A research agenda. In *Proceedings of the 2019 ACM SIGIR International Conference on Theory of Information Retrieval*, pages 217–220. ACM.

Shany Barhom, Vered Shwartz, Alon Eirew, Michael Bugert, Nils Reimers, and Ido Dagan. 2019. Revisiting joint modeling of cross-document entity and event coreference resolution. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 4179–4189, Florence, Italy. Association for Computational Linguistics.

Olivier Bodenreider. 2004. The unified medical language system (umls): Integrating biomedical terminology. *Nucleic acids research*, 32.

Xiao Cheng and Dan Roth. 2013. Relational inference for wikification. In *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing*, pages 1787–1796.

Silviu Cucerzan. 2007. Large-scale named entity disambiguation based on wikipedia data. In *Proceedings of the 2007 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning (EMNLP-CoNLL)*, pages 708–716.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4171–4186, Minneapolis, Minnesota. Association for Computational Linguistics.

Bhuwan Dhingra, Manzil Zaheer, Vidhisha Balachandran, Graham Neubig, Ruslan Salakhutdinov, and William W Cohen. 2020. Differentiable reasoning over a virtual knowledge base. In *International Conference on Learning Representations (ICLR)*.

Greg Durrett and Dan Klein. 2014. A joint model for entity analysis: Coreference, typing, and linking. *Transactions of the Association for Computational Linguistics*, 2:477–490.

Sourav Dutta and Gerhard Weikum. 2015. C3EL: A joint model for cross-document co-reference resolution and entity linking. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 846–856, Lisbon, Portugal. Association for Computational Linguistics.

Octavian-Eugen Ganea and Thomas Hofmann. 2017. Deep joint entity disambiguation with local neural attention. *arXiv preprint arXiv:1704.04920*.

Johannes Hoffart, Mohamed Amir Yosef, Ilaria Bordino, Hagen Fürstenau, Manfred Pinkal, Marc Spaniol, Bilyana Taneva, Stefan Thater, and Gerhard Weikum. 2011. Robust disambiguation of named entities in text. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, pages 782–792. Association for Computational Linguistics.

Samuel Humeau, Kurt Shuster, Marie-Anne Lachaux, and Jason Weston. 2019. Poly-encoders: Transformer architectures and pre-training strategies for fast and accurate multi-sentence scoring. *CoRR abs/1905.01969. External Links: Link Cited by*, 2:2–2.

Diederik P Kingma and Jimmy Ba. 2014. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*.

Phong Le and Ivan Titov. 2018. Improving entity linking by modeling latent relations between mentions. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1595–1604.

Robert Leaman and Zhiyong Lu. 2016. Taggerone: joint named entity recognition and normalization with semi-markov models. *Bioinformatics*, 32(18):2839–2846.

Jinhyuk Lee, Wonjin Yoon, Sungdong Kim, Donghyeon Kim, Sunkyu Kim, Chan Ho So, and Jaewoo Kang. 2019. Biobert: a pre-trained biomedical language representation model for biomedical text mining. *CoRR*, abs/1901.08746.

Jiao Li, Yueping Sun, Robin J. Johnson, Daniela Sciaky, Chih-Hsuan Wei, Robert Leaman, Allan Peter Davis, Carolyn J. Mattingly, Thomas C. Wiegers, and Zhiyong Lu. 2016. BioCreative V CDR task corpus: a resource for chemical disease relation extraction. *Database*, 2016. Baw068.

Fangyu Liu, Ehsan Shareghi, Zaiqiao Meng, Marco Basaldella, and Nigel Collier. 2020. Self-alignment pre-training for biomedical entity representations.

Lajanugen Logeswaran, Ming-Wei Chang, Kenton Lee, Kristina Toutanova, Jacob Devlin, and Honglak Lee. 2019. Zero-shot entity linking by reading entity descriptions. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 3449–3460, Florence, Italy. Association for Computational Linguistics.

Christopher D. Manning, Mihai Surdeanu, John Bauer, Jenny Finkel, Steven J. Bethard, and David McClosky. 2014. The Stanford CoreNLP natural language processing toolkit. In *Association for Computational Linguistics (ACL) System Demonstrations*, pages 55–60.

Edgar Meij, Krisztian Balog, and Daan Odijk. 2014. Entity linking and retrieval for semantic search. In *Proceedings of the 7th ACM International Conference on Web Search and Data Mining*, WSDM '14, pages 683–684, New York, NY, USA. ACM.

Rada Mihalcea and Andras Csomai. 2007. Wikify!: linking documents to encyclopedic knowledge. In *Proceedings of the sixteenth ACM conference on Conference on information and knowledge management*, pages 233–242. ACM.

David Milne and Ian H Witten. 2008. Learning to link with wikipedia. In *Proceedings of the 17th ACM conference on Information and knowledge management*, pages 509–518. ACM.

Sunil Mohan and Donghui Li. 2019. Medmentions: A large biomedical corpus annotated with umls concepts. In *Automated Knowledge Base Construction*.

Shikhar Murty, Patrick Verga, Luke Vilnis, Irena Radovanovic, and Andrew McCallum. 2018. Hierarchical losses and new resources for fine-grained entity typing and linking. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 97–109, Melbourne, Australia. Association for Computational Linguistics.

Jonathan Raphael Raiman and Olivier Michel Raiman. 2018. Deeptype: multilingual entity linking by neural type system evolution. In *Thirty-Second AAAI Conference on Artificial Intelligence*.

Lev Ratinov, Dan Roth, Doug Downey, and Mike Anderson. 2011. Local and global algorithms for disambiguation to wikipedia. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies-Volume 1*, pages 1375–1384. Association for Computational Linguistics.

Sebastian Riedel, Limin Yao, Andrew McCallum, and Benjamin M Marlin. 2013. Relation extraction with matrix factorization and universal schemas. In *Proceedings of the 2013 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 74–84.

Sunghwan Sohn, Donald C. Comeau, Won Gu Kim, and W. John Wilbur. 2008. Abbreviation definition identification based on automatic precision estimates. *BMC Bioinformatics*, 9:402.

Mujeen Sung, Hwisang Jeon, Jinhyuk Lee, and Jaewoo Kang. 2020. Biomedical entity representations with synonym marginalization. *arXiv preprint arXiv:2005.00239*.

Mihai Surdeanu, Julie Tibshirani, Ramesh Nallapati, and Christopher D Manning. 2012. Multi-instance multi-label learning for relation extraction. In *Proceedings of the 2012 joint conference on empirical methods in natural language processing and computational natural language learning*, pages 455–465. Association for Computational Linguistics.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *Advances in neural information processing systems*, pages 5998–6008.

Jesse Vig and Kalai Ramea. 2019. Comparison of transfer-learning approaches for response selection in multi-turn conversations.

Thomas Wolf, Victor Sanh, Julien Chaumond, and Clement Delangue. 2019. Transfertransfo: A transfer learning approach for neural network based conversational agents. *arXiv preprint arXiv:1901.08149*.

Ledell Wu, Fabio Petroni, Martin Josifoski, Sebastian Riedel, and Luke Zettlemoyer. 2019. Zero-shot entity linking with dense entity retrieval. *arXiv preprint arXiv:1911.03814*.

Nishant Yadav, Ari Kobren, Nicholas Monath, and Andrew McCallum. 2019. Supervised hierarchical clustering with exponential linkage. In *International Conference on Machine Learning (ICML)*.

Ming Zhu, Busra Celikkaya, Parminder Bhatia, and Chandan K Reddy. 2019. Latte: Latent type modeling for biomedical entity linking. *arXiv preprint arXiv:1911.09787*.