Zeroshot Multimodal Named Entity Disambiguation for Noisy Social Media Posts

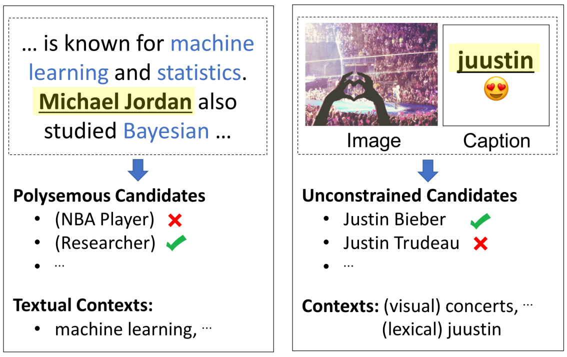
Seungwhan Moon[1,2], Leonardo Neves[2], Vitor Carvalho[3]
[1] Language Technologies Institute, Carnegie Mellon University
[2] Snap Research
[3] Intuit
seungwhm@cs.cmu.edu, lneves@snap.com, vitor_carvalho@intuit.com

Abstract

We introduce the new Multimodal Named Entity Disambiguation (MNED) task for multimodal social media posts such as Snapchat or Instagram captions, which are composed of short captions with accompanying images. Social media posts bring significant challenges for disambiguation tasks because 1) ambiguity not only comes from polysemous entities, but also from inconsistent or incomplete notations, 2) very limited context is provided with surrounding words, and 3) there are many emerging entities often unseen during training. To this end, we build a new dataset called *SnapCaptionsKB*, a collection of Snapchat image captions submitted to public and crowd-sourced stories, with named entity mentions fully annotated and linked to entities in an external knowledge base. We then build a deep zeroshot multimodal network for MNED that 1) extracts contexts from both text and image, and 2) predicts correct entity in the knowledge graph embeddings space, allowing for zeroshot disambiguation of entities unseen in training set as well. The proposed model significantly outperforms the state-of-the-art text-only NED models, showing efficacy and potentials of the MNED task.

1 Introduction

Online communications are increasingly becoming fast-paced and frequent, and hidden in these abundant user-generated social media posts are insights for understanding users and their preferences. However, these social media posts often come in unstructured text or images, making massive-scale opinion mining extremely challenging. Named entity disambiguation (NED), the task of linking ambiguous entities from free-form text *mention* to specific *entities* in a pre-defined knowledge base (KB), is thus a critical step for extracting structured information which leads to its application for recommendations, advertisement, personalized assistance, etc.



(a) Traditional NED (b) Multimodal NED

Figure 1: Examples of (a) a traditional NED task, focused on disambiguating polysemous entities based on surrounding textual contexts, and (b) the proposed Multimodal NED task for short media posts, which leverages both visual and textual contexts to disambiguate an entity. Note that mentions are often lexically inconsistent or incomplete, and thus a fixed candidates generation method (based on exact mention-entity statistics) is not viable.

While many previous approaches on NED been successful for well-formed text in disambiguating polysemous entities via context resolution, several additional challenges remain for disambiguating entities from extremely short and coarse text found in social media posts (*e.g.* "*juustin* 😍" as opposed to "*I love Justin Bieber/Justin Trudeau/etc.*"). In many of these cases it is simply impossible to disambiguate entities from text alone, due to enormous number of surface forms arising from incomplete and

Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Long Papers), pages 2000–2008
Melbourne, Australia, July 15 - 20, 2018. ©2018 Association for Computational Linguistics

inconsistent notations. In addition, social media posts often include mentions of newly emerging entities unseen in training sets, making traditional context-based entity linking often not viable.

However, as popular social media platforms are increasingly incorporating a mix of text and images (*e.g.* Snapchat, Instargram, Pinterest, etc.), we can advance the disambiguation task to incorporate additional visual context for understanding posts. For example, the mention of 'juuustin' is completely ambiguous in its textual form, but an accompanying snap image of a concert scene may help disambiguate or re-rank among several lexical candidates (*e.g.* Justin Bieber (a pop singer) versus Justin Trudeau (a politician) in Figure 1).

To this end, we introduce a new task called Multimodal Named Entity Disambiguation (MNED) that handles unique challenges for social media posts composed of extremely short text and images, aimed at disambiguationg entities by leveraging both textual and visual contexts.

We then propose a novel zeroshot MNED model, which obtains visual context vectors from images with a CNN (LeCun et al., 1989), and combines with textual context extracted from a bidirectional LSTM (Dyer et al., 2015) (Section 2.2). In addition, we obtain embeddings representation of 1M entities from a knowledge graph, and train the MNED network to predict label embeddings of entities in the same space as corresponding knowledge graph embeddings (Section 2.4). This approach effectively allows for zeroshot prediction of unseen entities, which is critical for scarce-label scenario due to extensive human annotation efforts required. Lastly, we develop a lexical embeddings model that determines lexical similarity between a mention and potential entities, to aid in prediction of a correct entity (Section 2.3). Section 2.5 details the model combining the components above.

Note that our method takes different perspectives from the previous work on NED (He et al., 2013; Yamada et al., 2016; Eshel et al., 2017) in the following important ways. First, while most of the previous methods generate fixed "candidates" for disambiguation given a mention from mention-entity pair statistics (thus disambiguation is limited for entities with exact surface form matches), we do not fixate candidate generation, due to intractable variety of surface forms for each named entity and unforeseen mentions of emerging entities. Instead, we have a lexical model incorporated into the discriminative score function that serves as soft normalization of various surface forms. Second, we extract auxiliary visual contexts for detected entities from user-generated images accompanied with textual posts, which is crucial because captions in our dataset are substantially shorter than text documents in most other NED datasets. To the best of our knowledge, our work is the first in using visual contexts for the named entity disambiguation task. See Section 4 for the detailed literature review.

Our contributions are as follows: for the new MNED task we introduce, we propose a deep zeroshot multimodal network with (1) a CNN-LSTM hybrid module that extracts contexts from both image and text, (2) a zeroshot learning layer which via embeddings projection allows for entity linking with 1M knowledge graph entities even for entities unseen from captions in training set, and (3) a lexical language model called *Deep Levenshtein* to compute lexical similarities between mentions and entities, relaxing the need for fixed candidates generation. We show that the proposed approaches successfully disambiguate incomplete mentions as well as polysemous entities, outperforming the state-of-the-art models on our newly crawled *SnapCaptionsKB* dataset, composed of 12K image-caption pairs with named entities annotated and linked with an external KB.

2 Proposed Methods

Figure 2 illustrates the proposed model, which maps each multimodal social media post data to one of the corresopnding entities in the KB. Given a multimodal input that contains a mention of an ambiguous entity, we first extract textual and visual features contexts with RCNNs and Bi-LSTMs, respectively (Section 2.2). We also obtain lexical character-level representation of a mention to compare with lexical representation of KB entities, using a proposed model called *Deep Levenshtein* (Section 2.3). We then get high-dimensional label embeddings of KB entities constructed from a knowledge graph, where similar entities are mapped as neighbors in the same space (Section 2.4). Finally, we aggregate all the contextual information extracted from surrounding text, image, and lexical notation of a mention, and predict the best matching KB entity based on knowledge graph label representation and lexical notation of KB entity candidates (Section 2.5).

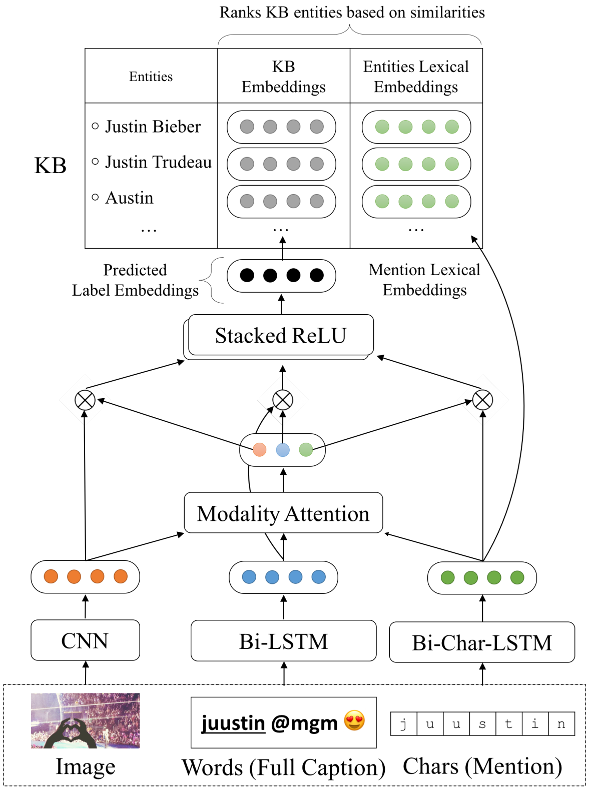


Figure 2: The main architecture of our Multimodal NED network. We extract contextual information from an image, surrounding words, and lexical embeddings of a mention. The modality attention module determines weights for modalities, the weighted projections of which produce label embeddings in the same space as knowledge-base (KB) entity embeddings. We predict a final candidate by ranking based on similarities with KB entity knowledge graph embeddings as well as with lexical embeddings.

2.1 Notations

Let $\mathbf{X} = \{\mathbf{x}^{(i)}\}_{i=1}^{N}$ a set of N input social media posts samples for disambiguation, with corresponding ground truth named entities $\mathbf{Y} = \{\mathbf{y}^{(i)}\}_{i=1}^{N}$ for $\mathbf{y} \in \mathbf{Y}_{\text{KB}}$, where $\mathbf{Y}_{\text{KB}}$ is a set of entities in KB. Each input sample is composed of three modalities: $\mathbf{x} = \{\mathbf{x}_w; \mathbf{x}_v; \mathbf{x}_c\}$, where $\mathbf{x}_w = \{\mathbf{x}_{w,t}\}_{t=1}^{L_w}$ is a sequence of words with length L_w surrounding a mention in a post, $\mathbf{x}_v$ is an image associated with a post (Section 2.2), and $\mathbf{x}_c = \{\mathbf{x}_{c,t}\}_{t=1}^{L_c}$ is a sequence of characters comprising a mention (Section 2.3), respectively. We denote high-dimensinal feature extractor functions for each modality as: $\mathbf{w}(\mathbf{x}_w)$, $\mathbf{c}(\mathbf{x}_c)$, $\mathbf{v}(\mathbf{x}_v)$. We represent each output label in two modalities: $\mathbf{y} = \{\mathbf{y}_{KB}; \mathbf{y}_c\}$, where $\mathbf{y}_{\text{KB}}$ is a knowledge base label embeddings representation (Section 2.4), and and $\mathbf{y}_c$ is a character embeddings representation of KB entities (Section 2.3: Deep Levenshtein).

We formulate our zeroshot multimodal NED task as follows:

$$\mathbf{y} = \operatorname*{argmax}_{\mathbf{y}' \in \mathbf{Y}_{\text{KB}}} \text{sim}\big(\mathbf{f}_{\mathbf{x}\rightarrow\mathbf{y}}(\mathbf{x}), \mathbf{y}'\big)$$

where $\mathbf{f}_{\mathbf{x}\rightarrow\mathbf{y}}$ is a function with learnable parameters that project multimodal input samples ($\mathbf{x}$) into the same space as label representations ($\mathbf{y}$), and $\text{sim}(\cdot)$ produces a similarity score between prediction and ground truth KB entities.

2.2 Textual and Visual Contexts Features

Textual features: we represent textual context of surrounding words of a mention with a Bi-LSTM language model (Dyer et al., 2015) with distributed word semantics embeddings. We use the following implementation for the LSTM.

$$\begin{aligned}
\mathbf{i}_t &= \sigma(\mathbf{W}_{xi}\mathbf{h}_{t-1} + \mathbf{W}_{ci}\mathbf{c}_{t-1}) \\
\mathbf{c}_t &= (1 - \mathbf{i}_t) \odot \mathbf{c}_{t-1} \\
&\quad + \mathbf{i}_t \odot \tanh(\mathbf{W}_{xc}\mathbf{x}_{w,t} + \mathbf{W}_{hc}\mathbf{h}_{t-1}) \\
\mathbf{o}_t &= \sigma(\mathbf{W}_{xo}\mathbf{x}_{w,t} + \mathbf{W}_{ho}\mathbf{h}_{t-1} + \mathbf{W}_{co}\mathbf{c}_t) \\
\mathbf{h}_t &= \mathbf{o}_t \odot \tanh(\mathbf{c}_t) \\
\mathbf{w}(\mathbf{x}_w) &= [\overrightarrow{\mathbf{h}_{L_w}}; \overleftarrow{\mathbf{h}_{L_w}}]
\end{aligned} \tag{1}$$

where $\mathbf{h}_t$ is an LSTM hidden layer output at decoding step t, and $\mathbf{w}(\mathbf{x}_w)$ is an output textual representation of bi-directional LSTM concatenating left and right context at the last decoding step $t = L_w$. Biase terms for gates are omitted for simplicity of formulation.

For the Bi-LSTM sentence encoder, we use pre-trained word embeddings obtained from an unsupervised language model aimed at learning co-occurrence statistics of words from a large external corpus. Word embeddings are thus represented as distributional semantics of words. In our experiments, we use pre-trained embeddings from Stanford GloVE model (Pennington et al., 2014).

Visaul features: we take the final activation of a modified version of the recurrent convolutional network model called Inception (GoogLeNet) (Szegedy et al., 2015) trained on the ImageNet dataset (Russakovsky et al., 2015) to classify multiple objects in the scene. The final layer representation ($\mathbf{v}(\mathbf{x}_v)$) thus encodes discriminative information describing what objects are shown in an image, providing cues for disambiguation.

2.3 Lexical Embeddings: Deep Levenshtein

While traditional NED tasks assume perfect lexical match between mentions and their corresopnding entities, in our task it is important to account for various surface forms of mentions (nicknames, mis-spellings, inconsistent notations, etc.) corresponding to each entity. Towards this goal, we train a separate deep neural network to compute approximate Levenshtein distance which we call Deep Levenshtein (Figure 3), composed of a shared bi-directional character LSTM, shared character embedding matrix, fully connected layers, and a dot product merge operation layer. The optimization is as follows:

$$\min_{\mathbf{c}} \left\| \frac{1}{2}\left(\frac{\mathbf{c}(\mathbf{x}_c) \cdot \mathbf{c}(\mathbf{x'}_c)^\top}{\|\mathbf{c}(\mathbf{x}_c)\|\|\mathbf{c}(\mathbf{x'}_c)\|} + 1 \right) - \text{sim}(\mathbf{x}_c, \mathbf{x'}_c) \right\|^2 \quad (2)$$

$$\text{where } \mathbf{c}(\mathbf{x}_c) = [\overrightarrow{\mathbf{h}_{c,L_c}}; \overleftarrow{\mathbf{h}_{c,L_c}}]$$

where $\mathbf{c}(\cdot)$ is a bi-directional LSTM output vector for a character sequence defined similar as in Eq.1, sim$(\cdot)$ is an output of the Deep Levenshtein network, producing a normalized similarity score with a range [0,1] based on Levenshtein edit distance, and $(\mathbf{x}_c, \mathbf{x'}_c)$ is any pair of two strings. We generate millions of these pairs as training data by artificially corrupting seed strings by varying degrees (addition, deletion, replacement).

Once trained, it can produce a purely lexical embedding of a string without semantic allusion (via $\mathbf{c}(\cdot)$), and predict lexical similarity between two strings based on their distance in the embedding space. On an intuitive level, this component effectively bypasses normalization steps, and instead incorporates lexical similarities between input mentions and output KB entities into the overall optimization of the disambiguation network.

We use by-product $\mathbf{c}(\cdot)$ network to extract lexical embedings of mentions and KB entities, and freeze $\mathbf{c}$ in training of the disambiguation network. We observe that this approach significantly outperforms alternative ways to obtain character embeddings (*e.g.* having a character Bi-LSTM as a part of the disambiguation network training, which unnecessarily learns semantic allusions that are prone to errors when notations are inconsistent.)

2.4 Label Embeddings from Knowledge Graph

Due to the overwhelming variety of (newly trending) entities mentioned over social media posts, at test phases we frequently encounter new named entities that are unseen in the training data. In order to address this issue, we propose a zeroshot learning approach (Frome et al., 2013) by inducing embeddings obtained from knowledge graphs on KB entities. Knowledge graph label embeddings are learned from known relations among entities within a graph (*e.g.* 'IS-A', 'LOCATED-AT', etc.), the resulting embeddings of which can group similar entities closer in the same space (*e.g.* 'pop stars' are in a small cluster, 'people' and 'organizations' clusters are far apart, etc.) (Bordes et al., 2013; Wang et al., 2014; Nickel et al., 2016). Once high-level mapping from contextual information to label embeddings is learned, the knowledge-graph based zeroshot approach can improve the entity linking performance given ambiguous entities unseen in training data. In brief formulation, the model for obtaining embeddings from a knowledge graph (composed of subject-relation-object (s, r, o) triplets) is as follows:

$$P(\mathbb{I}_r(s,o) = 1 | \mathbf{e}, \mathbf{e}_r, \theta) = \text{score}_\theta\big(\mathbf{e}(s), \mathbf{e}_r(r), \mathbf{e}(o)\big) \quad (3)$$

where $\mathbb{I}_r$ is an indicator function of a known relation r for two entities (s,o) (1: valid relation, 0: unknown relation), $\mathbf{e}$ is a function that extracts embeddings for entities, $\mathbf{e}_r$ extracts embeddings for relations, and score$_\theta(\cdot)$ is a deep neural network that produces a likelihood of a valid triplet.

In our experiments, we use the 1M subset of the Freebase knowledge graph (Bast et al., 2014) to obtain label embeddings with the Holographic KB implementation by (Nickel et al., 2016).

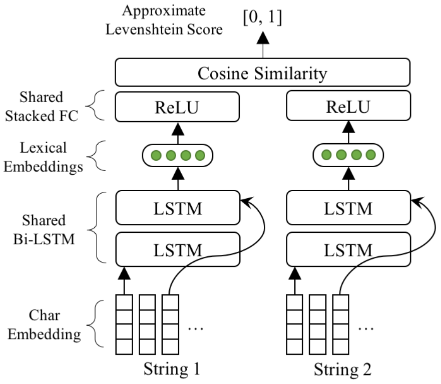


Figure 3: Deep Levenshtein, which predicts approximate Levenshtein scores between two strings. As a byproduct of this model, the shared Bi-LSTM can produce lexical embeddings purely based on lexical property of character sequences.

2.5 Deep Zeroshot MNED Network (DZMNED)

Using the contextual information extracted from surrounding text and an accompanying image (Section 2.2) and lexical embeddings of a mention (Section 2.3), we build a Deep Zeroshot MNED network (DZMNED) which predicts a corresponding KB entity based on its knowledge graph embeddings (Section 2.4) and lexical similarity (Section 2.3) with the following objective:

$$\min_{\mathbf{W}} \mathcal{L}_{\text{KB}}(\mathbf{x},\mathbf{y}_{\text{KB}};\mathbf{W_w},\mathbf{W_v},\mathbf{W_f})+\mathcal{L}_c(\mathbf{x}_c,\mathbf{y}_c;\mathbf{W_c})$$

where

$$\mathcal{L}_{\text{KB}}(\cdot)=$$

$$\frac{1}{N}\sum_{i=1}^{N}\sum_{\tilde{\mathbf{y}}\neq\mathbf{y}_{\text{KB}}^{(i)}}\max[0,\tilde{\mathbf{y}}\cdot\mathbf{y}_{\text{KB}}^{(i)}-\mathbf{f}(\overline{\mathbf{x}}^{(i)})\cdot(\mathbf{y}_{\text{KB}}^{(i)}-\tilde{\mathbf{y}})^{\top}]$$

$$\mathcal{L}_c(\cdot)=$$

$$\frac{1}{N}\sum_{i=1}^{N}\sum_{\tilde{\mathbf{y}}\neq\mathbf{y}_c^{(i)}}\max[0,\tilde{\mathbf{y}}\cdot\mathbf{y}_c^{(i)}-\mathbf{c}(\mathbf{x}_c^{(i)})\cdot(\mathbf{y}_c^{(i)}-\tilde{\mathbf{y}})^{\top}]$$

$\mathcal{R}(\mathbf{W})$: regularization

where $\mathcal{L}_{\text{KB}}(\cdot)$ is the supervised hinge rank loss for knowledge graph embeddings prediction, $\mathcal{L}_c(\cdot)$ is the loss for lexical mapping between mentions and KB entities, $\overline{\mathbf{x}}$ is a weighted average of three modalities $\mathbf{x}=\{\mathbf{x}_w;\mathbf{x}_v;\mathbf{x}_c\}$ via the modality attention module. $\mathbf{f}(\cdot)$ is a transformation function with stacked layers that projects weighted input to the KB embeddings space, $\tilde{\mathbf{y}}$ refers to the embeddings of negative samples randomly sampled from KB entities except the ground truth label of the instance, $\mathbf{W}=\{\mathbf{W_f},\mathbf{W_c},\mathbf{W_w},\mathbf{W_v}\}$ are the learnable parameters for $\mathbf{f}$, $\mathbf{c}$, $\mathbf{w}$, and $\mathbf{v}$ respectively, and $\mathcal{R}(\mathbf{W})$ is a weight decay regularization term.

Similarly to (Moon et al., 2018), we formulate the **modality attention** module for our MNED network as follows, which selectively attenuates or amplifies modalities:

$$[\mathbf{a}_w;\mathbf{a}_c;\mathbf{a}_v]=\sigma\big(\mathbf{W}_m\cdot[\mathbf{x}w;\mathbf{x}c;\mathbf{x}_v]+\mathbf{b}_m\big) \tag{4}$$

$$\alpha_m=\frac{\exp(\mathbf{a}_m)}{\sum_{m'\in\{w,c,v\}}\exp(\mathbf{a}_{m'})}\quad\forall m\in\{w,c,v\}$$

$$\overline{\mathbf{x}}=\sum_{m\in\{w,c,v\}}\alpha_m\mathbf{x}_m \tag{5}$$

where $\alpha=[\alpha_w;\alpha_c;\alpha_v]\in\mathbb{R}^3$ is an attention vector, and $\overline{\mathbf{x}}$ is a final context vector that maximizes information gain.

Intuitively, the model is trained to produce a higher dot product similarity between the projected embeddings with its correct label than with an incorrect negative label in both the knowledge graph label embeddings and the lexical embeddings spaces, where the margin is defined as the similarity between a ground truth sample and a negative sample.

At test time, the following label-producing nearest neighbor (1-NN) classifier is used for the target task (we cache all the label embeddings to avoid repetitive projections):

$$\text{1-NN}(\mathbf{x})=\operatorname*{argmax}_{(\mathbf{y}_{\text{KB}},\mathbf{y}_c)\in\mathbf{Y}_{\text{KB}}}\mathbf{f}(\overline{\mathbf{x}})\cdot\mathbf{y}_{\text{KB}}{}^{\top}+\mathbf{g}(\mathbf{x}_c)\cdot\mathbf{y_c}{}^{\top} \tag{6}$$

In summary, the model produces (1) projection of input modalities (mention, surrounding text, image) into the knowledge graph embeddings space, and (2) lexical embeddings representation of mention, which then calculates a combined score of contextual (knowledge graph) and string similarities with each entity in $\mathbf{Y}_{\text{KB}}$.

3 Empirical Evaluation

Task: Given a caption and an accompanying image (if available), the goal is to disambiguate and link a target mention in a caption to a corresponding entity from the knowledge base (1M subset of the Freebase knowledge graph (Bast et al., 2014)).

3.1 Datasets

Our **SnapCaptionsKB** dataset is composed of 12K user-generated image and textual caption pairs where named entities in captions and their links to KB entities are manually labeled by expert human annotators. These captions are collected exclusively from snaps submitted to public and crowd-sourced stories (aka *Live Stories* or *Our Stories*). Examples of such stories are "New York Story" or "Thanksgiving Story", which are aggregated collections of snaps for various public venues, events, etc. Our data do not contain raw images, and we only provide textual captions and obfuscated visual descriptor features extracted from the pre-trained InceptionNet. We split the dataset randomly into train (70%), validation (15%), and test sets (15%). The captions data have average length of 29.5 characters (5.57 words) with vocabulary size 16,553, where 6,803 are considered unknown tokens from Stanford GloVE embeddings (Pennington et al., 2014).

Named entities annotated in the dataset include many of new and emerging entities found in various surface forms. To the best of our knowledge, our *SnapCaptionsKB* is the only dataset that contains image-caption pairs with human-annotated named entities and their links to KB entities.

3.2 Baselines

We report performance of the following state-of-the-art NED models as baselines, with several candidate generation methods and variations of our proposed approach to examine contributions of each component (W: word, C: char, V: visual).

Candidates generation: Note that our zeroshot approach allows for entity disambiguation without a fixed candidates generation process. In fact, we observe that the conventional method for fixed candidates generation harms the performance for noisy social media posts with many emerging entities. This is because the difficulty of entity linking at test time rises not only from multiple entities (e) linking to a single mention (m), but also from each entity found in multiple surface forms of mentions (often unseen at train time). To show the efficacy of our approach that does not require candidates generation, we compare with the following candidates generation methods:

- $m \rightarrow e$ hash list: This method retrieves KB entity (e) candidates per mention (m) based on exact (m, e) pair occurrence statistics from a training corpora. This is the most predominantly used candidates generation method (He et al., 2013; Yamada et al., 2016; Eshel et al., 2017). Note that this approach is especially vulnerable at test time to noisy mentions or emerging entities with no or a few matching candidate entities from training set.

- k-NN: We also consider using lexical neighbors of mentions from KB entities as candidates. This approach can be seen as soft normalization to relax the issue of having to match a variety of surface forms of a mention to KB entities. We use our Deep Levenshtein (Section 2.3) to compute lexical embeddings of KB entities and mentions, and retrieves Euclidean neighbors (and their polysemous entities) as candidates.

NED models: We choose as baselines the following state-of-the-art NED models for noisy text, as well as several configurations of our proposed approach to examine contributions of each component (W: word, C: char, V: visual).

- sDA-NED (W only) (He et al., 2013): uses a deep neural network with stacked denoising autoencoders (sDA) to encode bag-of-words representation of textual contexts and to directly compare mentions and entities.

- ARNN (W only) (Eshel et al., 2017): uses an Attention RNN model that computes similarity between word and entity embeddings to disambiguate among fixed candidates.

- Deep Zeroshot (W only): uses the deep zeroshot architecture similar to Figure 2, but uses word contexts (caption) only.

- (**proposed**) DZMNED + Deep Levenshtein + InceptionNet with modality attention (W+C+V): is the proposed approach as described in Figure 2.

- (**proposed**) DZMNED + Deep Levenshtein + InceptionNet w/o modality attention (W+C+V): concatenates all the modality vectors instead.

- (**proposed**) DZMNED + Deep Levenshtein (W+C): only uses textual context.

- (**proposed**) DZMNED + Deep Levenshtein w/o modality attention (W+C): does not use the modality attention module, and instead concatenates word and lexical embeddings.

3.3 Results

Parameters: We tune the parameters of each model with the following search space (bold indicate the choice for our final model): character embeddings dimension: {25, 50, **100**, 150, 200, 300}, word embeddings size: {25, 50, **100**, 150, 200, 300}, knowledge graph embeddings size: {**100**, 200, 300}, LSTM hidden states: {50, **100**, 150, 200, 300}, and $\overline{x}$ dimension: {25, 50, **100**, 150, 200, 300}. We optimize the parameters with Adagrad (Duchi et al., 2011) with batch size 10, learning rate 0.01, epsilon 10^{-8}, and decay 0.1.

Main Results: Table 1 shows the Top-1, 3, 5, 10, and 50 candidates retrieval accuracy results on the *Snap Captions* dataset. We see that the proposed approach significantly outperforms the baselines which use fixed candidates generation

Modalities	Model	Candidates Generation	Accuracy (%)				
			Top-1	Top-3	Top-5	Top-10	Top-50
W	ARNN (Eshel et al., 2017)	$m{\rightarrow}e$ list	51.2	60.4	66.5	66.9	66.9
W	ARNN	5-NN (lexical)	35.2	43.3	45.0	-	-
W	ARNN	10-NN (lexical)	31.9	40.1	44.5	50.7	-
W	sDA-NED (He et al., 2013)	$m{\rightarrow}e$ list	48.7	57.3	66.3	66.9	66.9
W	Zeroshot	N/A	43.6	63.8	67.1	70.5	77.2
W + C	DZMNED	N/A	67.0	72.7	74.8	76.8	85.0
W + C	DZMNED + Modality Attention	N/A	67.8	73.5	74.8	76.2	84.6
W + C + V	DZMNED	N/A	67.2	74.6	77.7	80.5	**88.1**
W + C + V	DZMNED + Modality Attention	N/A	**68.1**	**75.5**	**78.2**	**80.9**	87.9

Table 1: NED performance on the *SnapCaptionsKB* dataset at Top-1, 3, 5, 10, 50 accuracies. The classification is over 1M entities. Candidates generation methods: N/A, or over a fixed number of candidates generated with methods: $m{\rightarrow}e$ hash list and kNN (lexical neighbors).

KB Embeddings	Top-1	Top-5	Top-10
Trained with 1M entities	**68.1**	**78.2**	**80.9**
Trained with 10K entities	60.3	72.5	75.9
Random embeddings	41.4	45.8	48.0

Table 2: MNED performance (Top-1, 5, 10 accuracies) on SnapCaptionsKB with varying qualities of KB embeddings. Model: DZMNED (W+C+V)

method. Note that $m \rightarrow e$ hash list-based methods, which retrieve as candidates the KB entities that appear in the training set of captions only, has upper performance limit at 66.9%, showing the limitance of fixed candidates generation method for unseen entities in social media posts. k-NN methods which retrieve lexical neighbors of mention (in an attempt to perform soft normalization on mentions) also do not perform well. Our proposed zeroshot approaches, however, do not fixate candidate generation, and instead compares combined contextual and lexical similarities among all 1M KB entities, achieving higher upper performance limit (Top-50 retrieval accuracy reaches 88.1%). This result indicates that the proposed zeroshot model is capable of predicting for unseen entities as well. The lexical sub-model can also be interpreted as functioning as soft neural mapping of mention to potential candidates, rather than heuristic matching to fixed candidates.

In addition, when visual context is available (W+C+V), the performance generally improves over the textual models (W+C), showing that visual information can provide additional contexts for disambiguation. The modality attention module also adds performance gain by re-weighting the modalities based on their informativeness.

Error Analysis: Table 3 shows example cases where incorporation of visual contexts affects disambiguation of mentions in textual captions. For example, polysemous entities such as 'Jordan' in the caption "*Taking the new Jordan for a walk*" or 'CID' as in "*LETS GO CID*" are hard to disambiguate due to the limited textual contexts provided, while visual information (*e.g.* visual tags 'footwear' for Jordan, 'DJ' for CID) provides similarities to each mention's distributional semantics from other training examples. Mentions unseen at train time ('STEPHHHH', 'murica') often resort to lexical neighbors by (W+C), whereas visual contexts can help disambiguate better. A few cases where visual contexts are not helpful include visual tags that are not related to mentions, or do not complement already ambiguous contexts.

Sensitivity to KB Embeddings Quality: The proposed approach relies its prediction on entity matching in the KB embeddings space, and hence the quality of KB embeddings is crucial for successful disambiguation. To characterize this aspect, we provide Table 2 which shows MNED performance with varying quality of embeddings as follows: KB embeddings learned from 1M knowledge graph entities (same as in the main experiments), from 10K subset of entities (less triplets to train with in Eq.3, hence lower quality), and random embeddings (poorest) - while all the other parameters are kept the same. It can be seen that the performance notably drops with lower quality of KB embeddings. When KB embeddings are replaced by random embeddings, the network effectively prevents the contextual zeroshot matching to KB entities and relies only on lexical similarities, achieving the poorest performance.

	Caption (target)	Visual Tags	GT	Top-1 Prediction	
				(W+C+V)	(W+C)
+	*“YA BOI STEPHHHH”*	sports equip, ball, parade, ...	Stephen Curry	(=GT)	Stephenville
	“Taking the new Jordan for a walk”	footwear, shoe, sock, ...	Air Jordan	(=GT)	Michael Jordan
	“out for murica’s bday 😎*”*	parade, flag, people, ...	U.S.A.	(=GT)	Murcia (Spain)
	“Come on now, Dre”	club, DJ, night, ...	Dr. Dre	(=GT)	Dre Kirkpatrick
	“LETS GO CID”	drum, DJ, drummer, ...	CID (DJ)	(=GT)	CID (ORG)
-	*“kick back hmu for addy.”*	weather, fog, tile, ...	Adderall	GoDaddy	(=GT)
	“@Sox to see 3️⃣4️⃣ *get retired!*⚾🍺*”*	sunglasses, stadium, ...	Red Sox	White Sox	White Sox

Table 3: Error analysis: **when do images help NED**? Ground-truth (GT) and predictions of our model with vision input (W+C+V) and the one without (W+C) for the underlined mention are shown. For interpretability, visual tags (label output of InceptionNet) are presented instead of actual feature vectors.

4 Related Work

NED task: Most of the previous NED models leverage local textual information (He et al., 2013; Eshel et al., 2017) and/or document-wise global contexts (Hoffart et al., 2011; Chisholm and Hachey, 2015; Pershina et al., 2015; Globerson et al., 2016), in addition to other auxiliary contexts or priors for disambiguating a mention. Note that most of the NED datasets (*e.g.* TAC KBP (Ji et al., 2010), ACE (Bentivogli et al., 2010), CoNLL-YAGO (Hoffart et al., 2011), etc.) are extracted from standardized documents with web links such as Wikipedia (with relatively ample textual contexts), and that named entitiy disambiguation specifically for short and noisy social media posts are rarely discussed. Note also that most of the previous literature assume the availability of “candidates” or web links for disambiguation via mention-entity pair counts from training set, which is vulnerable to inconsistent surface forms of entities predominant in social media posts.

Our model improves upon the state-of-the-art NED models in three very critical ways: (1) incorporation of visual contexts, (2) addition of the zeroshot learning layer, which allows for disambiguation of unseen entities during training, and (3) addition of the lexical model that computes lexical similarity entities to correctly recognize inconsistent surface forms of entities.

Multimodal learning studies learning of a joint model that leverages contextual information from multiple modalities in parallel. Some of the relevant multimodal learning task to our MNED system include the multimodal named entity recognition task (Moon et al., 2018), which leverages both text and image to classify each token in a sentence to named entity or not. In their work, they employ an entity LSTM that takes as input each modality, and a softmax layer that outputs an entity label at each decoding step. Contrast to their work, our MNED addresses unique challenges characterized by zeroshot ranking of 1M knowledge-base entities (vs. categorical entity types prediction), incorporation of an external knowledge graph, lexical embeddings, etc. Another is the multimodal machine translation task (Elliott et al., 2015; Specia et al., 2016), which takes as input text in source language as well as an accompanying image to output a translated text in target language. These models usually employ a sequence-to-sequence architecture (*e.g.* target language decoder takes as input both encoded source language and images) often with traditional attention modules widely used in other image captioning systems (Xu et al., 2015; Sukhbaatar et al., 2015). To the best of our knowledge, our approach is the first multimodal learning work at incorporating visual contexts for the NED task.

5 Conclusions

We introduce a new task called Multimodal Named Entity Disambiguation (MNED), which is applied on short user-generated social media posts that are composed of text and accompanying images. Our proposed MNED model improves upon the state-of-the-art models by 1) extracting visual contexts complementary to textual contexts, 2) by leveraging lexical embeddings into entity matching which accounts for various surface forms of entities, removing the need for fixed candidates generation process, and 3) by performing entity matching in the distributed knowledge graph embeddings space, allowing for matching of unseen mentions and entities by context resolutions.

References

Hannah Bast, Florian Baurle, Bjorn Buchhold, and Elmar Haussmann. 2014. Easy access to the freebase dataset. In *WWW*.

Luisa Bentivogli, Pamela Forner, Claudio Giuliano, Alessandro Marchetti, Emanuele Pianta, and Kateryna Tymoshenko. 2010. Extending english ace 2005 corpus annotation with ground-truth links to wikipedia. In *Proceedings of the 2nd Workshop on The Peoples Web Meets NLP: Collaboratively Constructed Semantic Resources*, pages 19–27.

Antoine Bordes, Nicolas Usunier, Alberto Garcia-Duran, Jason Weston, and Oksana Yakhnenko. 2013. Translating embeddings for modeling multi-relational data. In *NIPS*, pages 2787–2795.

Andrew Chisholm and Ben Hachey. 2015. Entity disambiguation with web links. *Transactions of the Association of Computational Linguistics*, 3(1):145–156.

John Duchi, Elad Hazan, and Yoram Singer. 2011. Adaptive subgradient methods for online learning and stochastic optimization. *JMLR*.

Chris Dyer, Miguel Ballesteros, Wang Ling, Austin Matthews, and Noah A Smith. 2015. Transition-based dependency parsing with stack long short-term memory. *ACL*.

Desmond Elliott, Stella Frank, and Eva Hasler. 2015. Multi-language image description with neural sequence models. *CoRR, abs/1510.04709*.

Yotam Eshel, Noam Cohen, Kira Radinsky, Shaul Markovitch, Ikuda Yamada, and Omer Levy. 2017. Named entity disambiguation for noisy text. *CoNLL*.

Andrea Frome, Greg Corrado, Jon Shlens, Samy Bengio, Jeffrey Dean, Marc'Aurelio Ranzato, and Tomas Mikolov. 2013. Devise: A deep visual-semantic embedding model. In *NIPS*.

Amir Globerson, Nevena Lazic, Soumen Chakrabarti, Amarnag Subramanya, Michael Ringaard, and Fernando Pereira. 2016. Collective entity resolution with multi-focal attention. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, volume 1, pages 621–631.

Zhengyan He, Shujie Liu, Mu Li, Ming Zhou, Longkai Zhang, and Houfeng Wang. 2013. Learning entity representation for entity disambiguation.

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.

Heng Ji, Ralph Grishman, Hoa Trang Dang, Kira Griffitt, and Joe Ellis. 2010. Overview of the tac 2010 knowledge base population track. In *Third Text Analysis Conference (TAC 2010)*, volume 3, pages 3–13.

Yann LeCun, Bernhard Boser, John S Denker, Donnie Henderson, Richard E Howard, Wayne Hubbard, and Lawrence D Jackel. 1989. Backpropagation applied to handwritten zip code recognition. *Neural computation*.

Seungwhan Moon, Leonard Neves, and Vitor Carvalho. 2018. Multimodal named entity recognition for short social media posts. *NAACL*.

Maximilian Nickel, Lorenzo Rosasco, and Tomaso Poggio. 2016. Holographic embeddings of knowledge graphs. *AAAI*.

Jeffrey Pennington, Richard Socher, and Christopher D. Manning. 2014. Glove: Global vectors for word representation. In *EMNLP*.

Maria Pershina, Yifan He, and Ralph Grishman. 2015. Personalized page rank for named entity disambiguation. In *Proceedings of the 2015 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 238–243.

Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael Bernstein, Alexander C. Berg, and Li Fei-Fei. 2015. ImageNet Large Scale Visual Recognition Challenge. *IJCV*.

Lucia Specia, Stella Frank, Khalil Sima'an, and Desmond Elliott. 2016. A shared task on multimodal machine translation and crosslingual image description. In *WMT*, pages 543–553.

Sainbayar Sukhbaatar, Jason Weston, Rob Fergus, et al. 2015. End-to-end memory networks. In *NIPS*, pages 2440–2448.

C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. E. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, and A. Rabinovich. 2015. Going deeper with convolutions. *CVPR*.

Zhen Wang, Jianwen Zhang, Jianlin Feng, and Zheng Chen. 2014. Knowledge graph embedding by translating on hyperplanes. In *AAAI*, pages 1112–1119. Citeseer.

Kelvin Xu, Jimmy Ba, Ryan Kiros, Kyunghyun Cho, Aaron Courville, Ruslan Salakhutdinov, Richard S Zemel, and Yoshua Bengio. 2015. Show, attend and tell: Neural image caption generation with visual attention. *arXiv preprint arXiv:1502.03044*, 2(3):5.

Ikuya Yamada, Hiroyuki Shindo, Hideaki Takeda, and Yoshiyasu Takefuji. 2016. Joint learning of the embedding of words and entities for named entity disambiguation. *CoNLL*.