Use FOFE-based Deep Neural Networks for Entity Discovery and Linking

Mingbin Xu, Nargiza Nosirova, Kelvin Jiang, Feng Wei, Hui Jiang
Department of Electrical Engineering and Computer Science
York University, 4700 Keele Street, Toronto, Ontario, M3J 1P3, Canada
{xmb,nana,kelvin,fwei,hj}@eecs.yorku.ca

Abstract

In this paper, we have studied a novel deep learning method for both mention detection and Entity Linking. Our methods rely on the recent fixed-size ordinally forgetting encoding (FOFE) method to fully encode each sentence fragment and its left/right contexts into a fixed-size representation. Afterwards, a simple feedforward neural network is used to reject or predict entity label for each individual fragment for mention detection and to compute similarity scores with Freebase nodes in the candidate list for entity linking. Experimental results in the 2017 KBP trilingual EDL track have shown that these methods have achieved strong results in both mention detection and entity linking.

1 Introduction

In this paper, we describe the main techniques used to build our entity discovery and linking system for the KBP2017 trilingual entity discovery and linking (EDL) track. The EDL task requires to detect named entities of five different types (PER, LOC, ORG, GPE, FAC) and their nominal mentions in the raw text of three languages (English, Chinese and Spanish) and further link each detected mention to the corresponding node in an existing knowledge base, namely Freebase. For NIL mentions that do not exist in the knowledge base, the EDL system needs to cluster all NIL mentions and assign a unique ID to each NIL mention cluster.

In this paper, we have studied a novel deep learning model for both mention detection (MD) and Entity Linking. Our methods rely on the recent fixed-size ordinally forgetting encoding (FOFE) method to fully any variable-length of text into a fixed-size representation without losing information. Experimental results in the 2017 KBP trilingual EDL track have shown that these methods have achieved the strong results in both mention detection and entity linking.

2 Preliminary: Fixed-size Ordinally Forgetting Encoding (FOFE)

In this section, we will first briefly review the FOFE technique used in our Entity Discovery and Linking (EDL) system.

Feedforward neural network (FNN) is a fast and powerful computation model. However, it requires to use the fixed-size inputs and lacks of the ability to capture long-term dependency in sequences. Because most NLP problems involves variable-length sequences of words, RNNs/LSTMs are more popular than regular feedforward NNs in dealing with these problems. The simple encoding method, called Fixed-size Ordinally Forgetting Encoding (FOFE), originally proposed in (Zhang et al., 2015b), nicely overcomes the limitations of deep FNNs because it can uniquely encode a variable-length sequence of words into a fixed-size representation without lossing information.

Give a vocabulary V consisting of $|V|$ distinct words, each word can be represented by a one-hot vector. FOFE mimics bag-of-words (BOW) but incorporates a forgetting factor to capture positional information. It encodes any sequence of variable length composed by words in V. Let $S = w_1, w_2, w_3, ..., w_T$ denote a sequence of T words from V, and $\boldsymbol{e_t}$ be the one-hot vector of the t-th word in S, where $1 \leq t \leq T$. The FOFE of each partial sequence $\boldsymbol{z_t}$ from the first word to the t-th word is recursively defined as:

$$\boldsymbol{z_t} = \begin{cases} \mathbf{0}, & \text{if } t = 0 \\ \alpha \cdot \boldsymbol{z_{t-1}} + \boldsymbol{e_t}, & \text{otherwise} \end{cases} \tag{1}$$

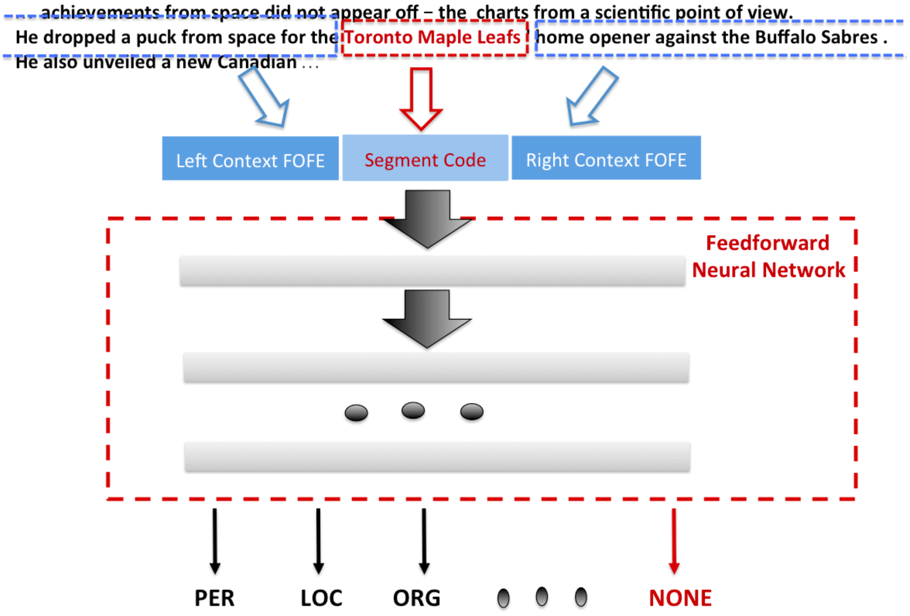
... achievements from space did not appear off – the charts from a scientific point of view.
He dropped a puck from space for the Toronto Maple Leafs home opener against the Buffalo Sabres .
He also unveiled a new Canadian ...
Left Context FOFE
Segment Code
Right Context FOFE
Feedforward Neural Network
PER
LOC
ORG
NONE

Figure 1: Illustration of the local detection approach for NER using FOFE codes as input and a feedforward neural network as model. The window currently examines the fragment of *Toronto Maple Leafs*. The window will scan and scrutinize all fragments up to K words.

where the constant α is called forgetting factor, and it is chosen picked between 0 and 1 exclusively. Obviously, the size of z_t is $|V|$, and it is irrelevant to the length of original sequence, T. According to (Zhang et al., 2015b), FOFE is capable of uniquely encoding any sequence of arbitrary length, serving as a fixed-size but theoretically lossless representation for any sequence.

3 Entity Discovery and Mention Detection

In this work, we have used a new FOFE-based local detection approach to build our systems for entity discovery and mention detection, and then we have studied how to use multi-style training and model ensemble to further improve the performance.

3.1 FOFE-based Models for Named Entity Recognition and Mention Detection

Our systems, called **FOFE-NER** hereafter, are motivated by the way how human actually infers whether a word segment in text is an entity or mention. To a large extent, the meaning and spelling of the underlying fragment are informative to distinguish named entities from the rest of the text. Contexts play a very important role in NER or mention detection when it involves multi-sense words/phrases or out-of-vocabulary (OOV) words. As shown in Figure 1, our proposed **FOFE-NER** method will examine all possible fragments in text (up to a certain length) one by one. For each fragment, it uses the FOFE method to fully encode the underlying fragment itself, its left context and right context into some fixed-size representations, which are in turn fed to a multi-layer feedforward neural network to predict whether the current fragment is not a valid entity mention (*NONE*), or its correct entity type (*PER*, *LOC*, *ORG* and so on). This method is appealing because the FOFE codes serves as a theoretically lossless representation of the hypothesis and its full contexts and the multi-layer neural networks are used as a universal approximator to map from text to the entity labels.

In this work, we use FOFE to explore both word-level and character-level features for each fragment and its contexts. **FOFE-NER** generates several word-level features for each fragment hypothesis and its left and right contexts: i) Bag-of-word vector of the fragment; ii) FOFE code for left context including the fragment; iii) FOFE code for left context excluding the fragment; iv) FOFE code for right context including the fragment; v) FOFE code for right context excluding the fragment. On top of the above word-level features, we also augment character-level features for the underlying segment hypothesis to further model its morphological structure: i) Left-to-right FOFE code of the character sequence of the underlying fragment; ii) Right-to-left FOFE code of the character sequence of the underlying fragment.

Obviously, the above **FOFE-NER** model will take each sentence of words, $S = [w_1, w_2, w_3, ..., w_m]$, as input, and examine all continuous sub-sequences $[w_i, w_{i+1}, w_{i+2}, ..., w_j]$ up to n words in S for possible entity types. All sub-sequence longer than n words are considered as non-entity in this work.

When we train the model, based on the entity labels of all sentences in the training set, we will generate many sentence fragments up to K words. These fragments fall into three categories: i) Exact-match with an entity label, e.g., the fragment "*Toronto Maple Leafs*" in the previous example; ii) Partial-overlap with an entity label, e.g., "*for the Toronto*"; iii) Disjoint with all entity label, e.g. "*from space for*". For all exact-matched fragments, we generate the corresponding outputs based on the types of the matched entities in the training set. For both partial-overlap and disjoint fragments, we introduce a new output label, **NONE**, to indicate that these fragments are not a valid entity. Therefore, the output nodes in the neural networks contains all entity types plus a rejection option denoted as **NONE**.

3.2 Multi-Task Learning

Recently, Multi-task Learning (MTL) has been the centre of a lot of attention. The main idea of MTL lies in concurrently learning a task alongside related (auxiliary) tasks by using a shared representation. Much of the work done in MTL was initiated by (Caruana, 1997). His definition of a related task is one that gives the main task a higher performance when trained together as opposed to training it on its own. He also proposes to share features and hidden modules for better performance. Such techniques have been used and confirmed in many studies (Maurer et al., 2016; Ando and Zhang, 2005). The wide spectrum of approaches in the MTL literature have yielded promising results that grant further investigation. To explore this, we develop a multi-task feedforward neural network model which relies on the recent fixed-size ordinally forgetting encoding (FOFE) to conduct entity discovery and mention detection.

The character and word features used for this model are FOFE-based, similarly to the single-task model mentioned in the previous section. The multi-task model consists of a feedforward neural network with hidden Rectified linear unit (ReLU) (Nair and Hinton, 2010) activation layers. The model makes use of multiple datasets with different named entity annotations, where each dataset is treated as a separate task. Assuming that we have access to N such tasks, a single task is randomly chosen out of the N tasks at the beginning of each epoch in order to train the neural network. The character and word features of the chosen task are fed into the shared module consisting of hidden layers, where the internal representations that arise in the hidden layers for one task may be used by other subsequent tasks. A fixed amount of training examples (mini-batch) from the chosen dataset are selected at each training step. As shown in Figure 2, the shared module then branches out to N private modules in order to also customize the learning of each individual task. Each private module contains an output layer that uses softmax activation, classifying the fragment by selecting the label with the highest value from the softmax output. Finally, we feed the loss of the current task backwards. Therefore, throughout the epochs, the inputs of each task get fed alternatively into the neural network. The layers are randomly initialized using a uniform distribution. Our model defines one of the tasks to be the main task (Task N in Figure 2); the task of which we want to improve the performance. Thus, the private module of the main task will contain a hidden layer in addition to an output layer, whereas the auxiliary private modules will simply consist of an output layer. After having chosen a task at the beginning of an epoch, the model will take each sequence of words as input from the data, and examine all subsequences in a sentence to up to n words for possible entity types. All subsequences longer than n words are considered as non-entities.

3.3 Model Ensemble

In this work, we adopt two word embedding approaches, i.e. word2vec (Mikolov et al., 2013) and fofe-embedding. Unlike (Sanu et al., 2017), 3 sets of fofe-embedding for each language are trained as a byproduct of FOFE-LM (Zhang et al., 2015b; Zhang et al., 2015a) bidirectionally on English Gigaword (Parker et al., 2011), Chinese Gigaword

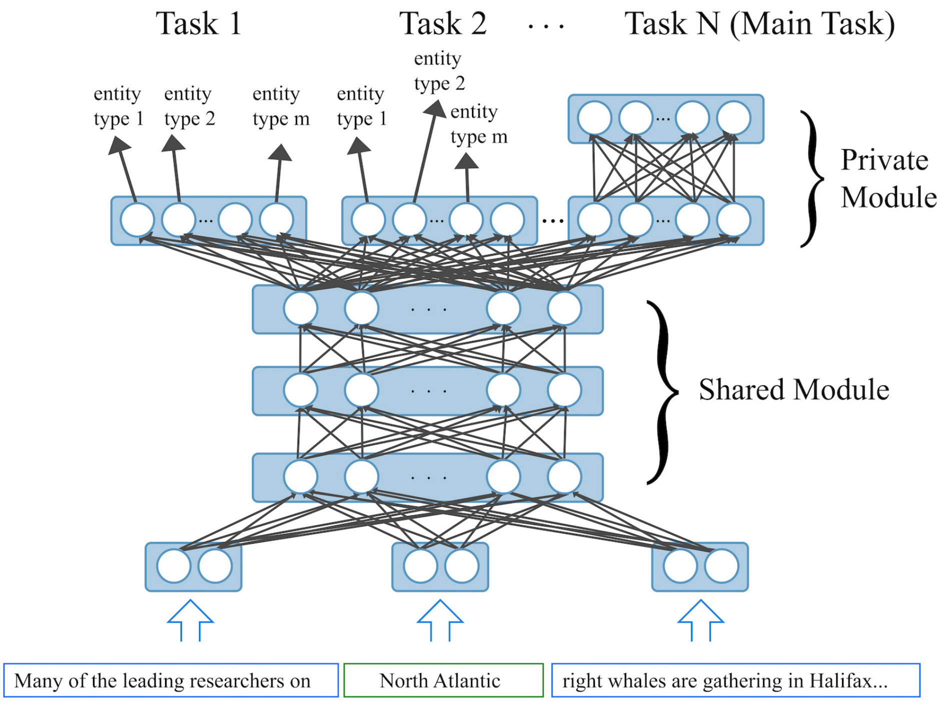

Figure 2: An illustration of the multi-task feedforward neural network approach for NER using FOFE codes as input. The window currently examines the fragment of *North Atlantic*.

(Graff and Chen, 2005) and Spanish Gigaword (Mendonca et al., 2009) respectively. We follow the hyper-parameters in (Zhang et al., 2015b; Zhang et al., 2015a) but speed it up by NCE (Gutmann and Hyvärinen, 2010). Word embeddings are extracted from the projection layer of FOFE-LM, where L_2 norm is applied.

4 Entity Linking and NIL Clustering

In the entity linking task, each detected mention needs to be linked to a known entity in an existing knowledge base, namely Freebase in this task. For all mentions that do not match any existing node in Freebase, we need to cluster these NIL mentions.

4.1 Entity Linking Baselines

Similar to (Liu et al., 2016), our first entity linking baseline system consists of two key modules: a candidate generation module and a neural networks ranking model. The candidate generation module is a complicated rule based system, which output a list of candidates for each detected mention. A simple multilayer feedforward neural networks model is trained to assign posterior probabilities to the candidates in the list based on some hand-crafted features (Liu et al., 2016). The candidate with the highest probability is chosen as the final linking result.

Our second linking system simply relies only on the detected mentions from a document. For each detected mention in a document, the multiple queries are sent to two online knowledge sources [1] using each underlying detected mention as well as some combinations with its neighbouring entities. The top 5 search results are recorded if they match an mid in Freebase. The final linking ID is determined based on a weighted voting from the all matched results.

4.2 FOFE Model for Entity Linking

Here, we have also investigated to use a novel approach for entity linking, which make use of feedforward NNs and the FOFE features (Zhang et al., 2015b) to fully encode the sentence fragment and its left/right contexts into a fixed-size representation, as in (Xu et al., 2017). Comparing with our baseline system in (Liu et al., 2016), this method does not use any feature engineering and totally relies on the FOFE features that are automatically extracted from text.

As we know, the final linking performance heavily relies on the generated candidate list. In this work, we utilize a rule-based system, simi-

[1] www.google.com and www.wikipedia.org

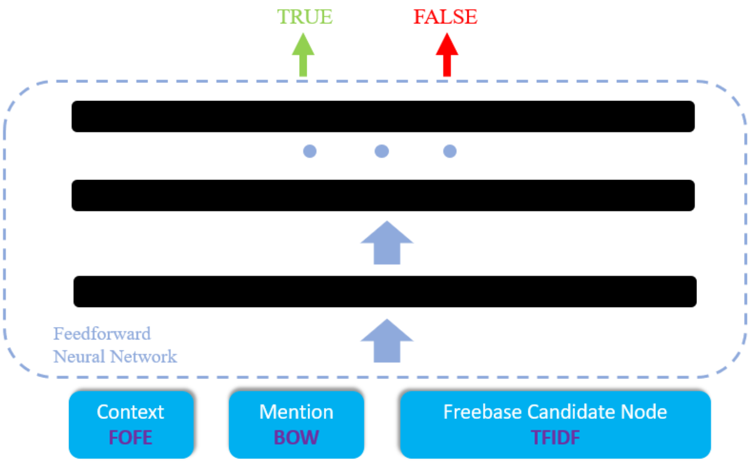

Figure 3: Our Entity Linking system use FOFE features as input and a feed-forward neural network as model.

lar to (Liu et al., 2016), as our candidate generation module to generate candidates for each detected mention. Candidates are generated based on knowledge bases, including Freebase and Wikipedia. The output of this module is a candidate list, consisting of a list of Freebase nodes possibly matching this mention.

The candidate list for each detected mention contains the Freebase node IDs that match with the mention in the candidate generation process. In this work, we propose to use a feedforward fully connected NNs ranking model to assign probabilities to all candidates in the list. The candidate with the highest probability is chosen as the final linking result. Each time, our FOFE ranking model takes the mention and a candidate from the list to compute a score, as shown in Figure 3. For each pair of the detected mention and a candidate Freebase node, we use FOFE and bag-of-words to generate fixed-size features.

4.3 Nil Entity Clustering

For all mentions identified as NIL by the above NN ranking models, we perform a very simple rule-based algorithm to cluster them: Different named NIL mentions are grouped into one cluster only if their mention strings are the same (case-insensitive).

5 Experimental Results

The trilingual EDL task is extended to to detect nominal mentions of all 5 entity types for all three languages. In our experiments, for simplicity, we just treat nominal mention types as some extra entity types and detect them along with named entities together with a single model. In the following, we will report the our performance on the KBP2017 trilingual EDL track.

5.1 Training Data

We make use of the following data sets as our training data to learn the NER and mention detection models.

- **Training and evaluation data in KBP2015**: In previous year's competition, 335 English documents, 313 Chinese documents and 296 Spanish documents were annotated for training and evaluation, totalling 944 documents. In this data set, all five named mention types (PER, ORG, GPE, LOC, FAC) and only one nominal mention type (PER) are labelled.
- **Evaluation data in KBP2016**: In this data set, all five named mention types (PER, ORG, GPE, LOC, FAC) and their nominal mention type (PER) are labelled.
- **iFLYTEK's in-house dataset**: The iFLYTEK Research has generously shared with us about 10,000 in-house English and Chinese labeled documents (Liu et al., 2016). These documents are internally labelled by iFLYTEK using some annotation rules similar to the KBP 2016 guidelines.

5.2 Multi-task Learning Results

In our experiments, we treat the KBP EDL task as the main task for all three languages. The

KBP2015 and KBP2016 datasets are used as training data for the KBP EDL task. For each language, we set up three multi-task models that are trained and evaluated independently.

For the English model, we make use of the CoNLL-2003 and OntoNotes 5.0 datasets as auxiliary tasks. The CoNLL-2003 dataset (Tjong Kim Sang and De Meulder, 2003) consists of newswire data originated from the Reuters RCV1 corpus. It is tagged with four entity types: person (PER), location (LOC), organization (ORG) and miscellaneous (MISC). The OntoNotes 5.0 dataset consists of text from sources such as broadcast conversation and news, newswire, telephone conversation, magazine and web text. This dataset was assembled by (Pradhan et al., 2013) for the CoNLL-2012 shared task, who specifies a standard train, validation, and test split followed in our evaluation. It is tagged with eighteen entity types, some of which are: person, facility, organization, product, data, time, money, quantity and so forth. The model has two hidden layers in the shared module and one hidden layer in the main tasks private module. The hidden layers each contain 512 units and are trained with the Stochastic Gradient Descent optimizer using mini-batch of size 256 for 128 epochs. The only form of regularization used is dropout (Srivastava et al., 2014) with a probability of 0.5. A learning rate is set to 0.128 with a decay factor of 1/16 if the validation performance drops. The loss function used is categorical cross entropy. We set the forgetting factor to $\alpha = 0.5$. We use three sets of word embeddings of 256 dimensions from the English, Spanish and Chinese gigawords for the three models. The Spanish and Chinese models only utilize the DAFT Light ERE dataset as the auxiliary task. The DEFT Light ERE dataset consists of discussion forum and newswire documents tagged with five types of named entities: person, title, organization, geopolitical entities and location. The setup for both languages is similar to the English model, except the learning rate used is 0.064, with mini-batch of size 128. Also, the Spanish model contains a single hidden layer in the shared module. We combine our inhouse dataset to KBP2017 training data for the English and Chinese models.

We have evaluated the above multi-task learning (MTL) method for KBP2016 evaluation data set and significant gains are observed for all three languages. The final (MTL) performance in the KBP2017 official evaluation is shown in Table 1.

5.3 Model Ensemble Results

We have trained 4 sets of **FOFE-NER** models (Xu et al., 2017; Xu and Jiang, 2016; Xu et al., 2016). They differ in terms of initial word embedding and training data, as listed in Table 2. Each set consists of 5 models that are cross-validated, which totals 20 models. For each model set, data is divided into 5 disjoint partitions. Each model within the same set holds out one different partition and is trained with the rest.

Final result is produced by hard voting where 6 models agree (Breiman, 1996). More precisely, we treat the labels as 4-element tuples of (filename, offsets, entity-type, named-or-nominal). Occurrences of each label are accumulated while each occurrence contributes to 1 voting score. If a label receives more than 6 votes out of 20, it is included in the final submission. As shown in Table 3, the model ensemble has significantly improved the entity discovery (ED) performance for all three languages, about 1.5% gain in F_1 scores.

5.4 Entity Linking Results

We have compared the FOFE-based entity linker with our two baseline systems described above. The performance comparison is shown in Table 4. The proposed FOFE-based entity linker has achieved similar performance as our two baseline systems, one using handcrafted features and the other using online search engines.

6 Conclusions

In this paper, we have described the techniques used for KBP2017 evaluation of Trilingual EDL Track. We have investigated a new FOFE-based local detection based approach for both entity discovery and linking. This method has relied on the recent fixed-size ordinally forgetting encoding (FOFE) method to fully encode each fragment and its left/right contexts into a fixed-size representation, and a simple feedforward neural network to reject or predict entity label for each individual fragment. Our systems using this method have achieved a strong performance in the official KBP2017 trilingual EDL evaluations.

References

Rie Kubota Ando and Tong Zhang. 2005. A framework for learning predictive structures from multi-

LANG	Training data	Auxiliary tasks	F_1
ENG	KBP2015 (English) + in-house	CoNLL2003, OntoNotes 5.0	0.780
CMN	KBP2015 (Chinese) + in-house	DEFT Light ERE Chinese	0.685
SPA	KBP2015 (Spanish)	DEFT Light ERE Spanish	0.767

Table 1: Summary of the training data and auxiliary tasks used for the entity discovery performance in KBP2017 EDL evaluation.

	Embedding	Training Data
set1	fofe-embedding	KBP2015, KBP2016, in-house
set2	fofe-embedding	KBP2016, in-house
set3	word2vec	KBP2015, KBP2016, in-house
set4	word2vec	KBP2016, in-house

Table 2: Four different sets of ED models are trained for ensemble.

LANG	single model			model ensemble		
	P	R	F_1	P	R	F_1
ENG	0.801	0.745	0.772	0.796	0.779	**0.787**
CMN	0.775	0.660	0.713	0.796	0.673	**0.729**
SPA	0.856	0.715	0.779	0.834	0.762	**0.797**
ALL	-	-	-	0.808	0.729	**0.767**

Table 3: Entity Discovery (ED) performance of model ensemble in the KBP 2017 trilingual EDL evaluation.

LANG	baseline1		baseline2		FOFE-EL	
	NERLC	CEAFmC	NERLC	CEAFmC	NERLC	CEAFmC
ENG	**0.646**	0.630	0.572	0.615	0.604	**0.646**
CMN	**0.617**	**0.650**	0.579	0.615	0.487	0.635
SPA	**0.569**	0.568	0.538	0.547	0.502	**0.572**
ALL	**0.611**	**0.607**	0.565	0.586	-	-

Table 4: Performance on the KBP2017 EDL evaluation of our three entity linking systems. (*NERLC* denotes for *strong_typed_all_match* and *CEAFmC* for *typed_mention_ceaf*)

ple tasks and unlabeled data. *J. Mach. Learn. Res.*, 6:1817–1853, December.

Leo Breiman. 1996. Bagging predictors. *Machine learning*, 24(2):123–140.

Rich Caruana. 1997. Multitask learning. *Mach. Learn.*, 28(1):41–75, July.

David Graff and Ke Chen. 2005. Chinese gigaword. *LDC Catalog No.: LDC2003T09, ISBN*, 1:58563–58230.

Michael Gutmann and Aapo Hyvärinen. 2010. Noise-contrastive estimation: A new estimation principle for unnormalized statistical models. In *Proceedings of the Thirteenth International Conference on Artificial Intelligence and Statistics*, pages 297–304.

Dan Liu, Wei Lin, Shiliang Zhang, Si Wei, and Hui Jiang. 2016. The USTC_NELSLIP systems for trilingual entity detection and linking tasks at TAC KBP 2016. In *Proceedings of Text Analysis Conference (TAC2016)*.

Andreas Maurer, Massimiliano Pontil, and Bernardino Romera-Paredes. 2016. The benefit of multitask representation learning. *J. Mach. Learn. Res.*, 17(1):2853–2884, January.

Angelo Mendonca, David Andrew Graff, and Denise DiPersio. 2009. *Spanish gigaword second edition*. Linguistic Data Consortium.

Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg S Corrado, and Jeff Dean. 2013. Distributed representations of words and phrases and their compositionality. In *Advances in neural information processing systems*, pages 3111–3119.

Vinod Nair and Geoffrey E. Hinton. 2010. Rectified linear units improve restricted boltzmann machines. In *Proceedings of the 27th International Conference on International Conference on Machine Learning*, ICML'10, pages 807–814, USA. Omnipress.

Robert Parker, David Graff, Junbo Kong, Ke Chen, and Kazuaki Maeda. 2011. English gigaword. *Linguistic Data Consortium*.

Sameer Pradhan, Alessandro Moschitti, Nianwen Xue, Hwee Tou Ng, Anders Bjrkelund, Olga Uryupina, Yuchen Zhang, and Zhi Zhong. 2013. Towards robust linguistic analysis using ontonotes. In *Proceedings of the Seventeenth Conference on Computational Natural Language Learning*, pages 143–152, Sofia, Bulgaria, August. Association for Computational Linguistics.

Joseph Sanu, Mingbin Xu, Hui Jiang, and Quan Liu. 2017. Word embeddings based on fixed-size ordinally forgetting encoding. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 310–315. Association for Computational Linguistics.

Nitish Srivastava, Geoffrey Hinton, Alex Krizhevsky, Ilya Sutskever, and Ruslan Salakhutdinov. 2014. Dropout: A simple way to prevent neural networks from overfitting. *J. Mach. Learn. Res.*, 15(1):1929–1958, January.

Erik F. Tjong Kim Sang and Fien De Meulder. 2003. Introduction to the conll-2003 shared task: Language-independent named entity recognition. In *Proceedings of the Seventh Conference on Natural Language Learning at HLT-NAACL 2003 - Volume 4*, CONLL '03, pages 142–147, Stroudsburg, PA, USA. Association for Computational Linguistics.

Mingbin Xu and Hui Jiang. 2016. A fofe-based local detection approach for named entity recognition and mention detection. *arXiv preprint arXiv:1611.00801*.

Mingbin Xu, Feng Wei, Sedtawut Watcharawittayakul, Yuchen Kang, and Hui Jiang. 2016. The yorknrm systems for trilingual edl tasks at tac kbp 2016. In *Proceedings of the 9th Text Analysis Conference (TAC2016)*.

Mingbin Xu, Hui Jiang, and Sedtawut Watcharawittayakul. 2017. A local detection approach for named entity recognition and mention detection. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics*. Association for Computational Linguistics (ACL).

Shiliang Zhang, Hui Jiang, Mingbin Xu, Junfeng Hou, and Lirong Dai. 2015a. A fixed-size encoding method for variable-length sequences with its application to neural network language models. *arXiv preprint arXiv:1505.01504*.

Shiliang Zhang, Hui Jiang, Mingbin Xu, Junfeng Hou, and Lirong Dai. 2015b. The fixed-size ordinally-forgetting encoding method for neural network language models. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 2: Short Papers)*. Association for Computational Linguistics (ACL).