Evaluating and Improving End-to-End Systems for Knowledge Base Population

Maxime Prieur, Cédric Du Mouza, Guillaume Gadek, Bruno Grilheres

Evaluating and Improving End-to-End Systems for Knowledge Base Population

Maxime Prieur[1], Cédric du Mouza[1], Guillaume Gadek[2] and Bruno Grilheres[2]
[1]*Cédric Laboratory, Conservatoire National des Arts et Métiers, Paris, France*
[2]*Airbus Defence and Space, Élancourt, France*
{*maxime.prieur.auditeur, cedric.dumouza*}*@lecnam.net, guillaume.gadek@airbus.com*

Keywords: Knowledge Base Population, Entity Linking, Supervised Learning, Data Mining, Method and Evaluation

Abstract: Knowledge Bases (KB) are used in many fields, such as business intelligence or user assistance. They aggregate knowledge that can be exploited by computers to help decision making by providing better visualization or predicting new relations. However, their building remains complex for an expert who has to extract and link each new information. In this paper, we describe an entity-centric method for evaluating an end-to-end Knowledge Base Population system. This evaluation is applied to ELROND, a complete system designed as a workflow composed of 4 modules (Named Entity Recognition, Coreference Resolution, Relation Extraction and Entity Linking) and MERIT, a dynamic entity linking model made of a textual encoder to retrieve similar entities and a classifier.

1 INTRODUCTION

Knowledge bases (KB) are data structures, generally relying on a predefined ontology, which are very useful for aggregating information in order to simplify its visualization and analysis. Knowledge bases are used in many fields such as business intelligence (Shue et al., 2009) to improve decision making or to extract elements linking scientific publications (Ammar et al., 2018). However, manual construction and updating of Knowledge Bases are extremely costly since the domains in which they are deployed usually exploit constantly evolving information. An alternative solution would be an automatic population that would extract and add the desired elements from sources of interest into the knowledge base.

Nevertheless, the existing solutions are not yet sufficient and this subject is still the focus of many workshops (Ghosal et al., 2022), which aim at proposing new approaches either on the end-to-end process workflow or on one of its subtasks. Most proposals on the literature focus on the optimization of one or two particular modules but not of an end-to-end system. However, we have no guarantee to get the best performances for an end-to-end system when trying to optimize each module separately. Unfortunately, to optimize an end-to-end Knowledge Base Population (KBP) system as a whole, there exists, to the best of our knowledge, no evaluation protocol that are both automatic and exhaustive (Min et al., 2018; Mesquita et al., 2019).

In this paper, we attempt to address this issue with the following contributions:

- we formalize a method for evaluating end-to-end Knowledge Base Population systems from texts as a whole;
- we present ELROND, an end-to-end system implemented as a 4-step processing workflow whick could be considered as a baseline for comparison with future solutions;
- as an improvement of ELROND baseline, we introduce MERIT, a textual encoder-based entity linking solution for entity resolution when building a dynamic knowledge base;
- we measure and compare the performances of the proposed models by following the presented evaluation method and applying it to the DWIE dataset. (Zaporojets et al., 2021).

The remaining of this article is structured as follows: after a presentation of recent work on knowledge base population (KBP) evaluation approaches, on end-to-end systems as well as on models for the entity linking task (Section 2), we formalize and detail our evaluation protocol (Section 3). Then we describe our end-to-end implementation proposal, ELROND, and our linking solution, MERIT (Section 4).

We present the experiments conducted and the results obtained in Section 5 before discussing the possible perspectives to explore in future work.

2 RELATED WORK

2.1 Knowledge Base Population

Knowledge Base Population from texts consists in extracting the elements and their relations of interest in order to add them to the already known and structured information. This task usually involves several steps: named entity recognition (NER), coreference resolution, relation extraction, and entity linking. TinkerBell (Al-Badrashiny et al., 2017), one of the first end-to-end systems, consists of a NER module, combining two Bi-LSTMs taking respectively the text and the linguistic features to tag the words in the document. Meanwhile, entity linking is solved through the sum of popularity, similarity and consistency scores. KnowledgeNet (Mesquita et al., 2019) also addresses relation extraction with a Bi-LSTM (Long Short-Term Memory) Huang et al. (2015) model which receives linguistic features and embeddings generated by a BERT (Bidirectional Encoder Representations from Transformers) model (Kenton and Toutanova, 2019). This model is however applied at the sentence level and discards supra-phrastic relations (Yao et al., 2019). Moreover these approaches are based on Wikipedia and are not adequate when the sources have a large proportion of entities not listed in the encyclopedia. KBPearl (Lin et al., 2020) suggests to use open information extraction (OIE) frameworks. The system extracts and links knowledge from the text using a graph densification method applied to the semantic graph built from the text. In addition to a profusion of potentially uninteresting information for the user caused by the lack of predicates in OIE frameworks, the selection of candidates is performed by alias matching, which is not very suitable when new mentions appear.

2.2 Entity Linking

Entity linking is the process of determining whether the mentions in a text refer to entities in the database. This process generally establishes a list of similar entities before selecting, or not, one of the candidates. In addition to dictionary-based approaches (Al-Badrashiny et al., 2017), encoder-based methods have been proposed. For instance BLINK (Wu et al., 2020), uses two BERT models to compare entity mentions and Wikipedia descriptions projected into a single representation space. This bi-encoder returns descriptions similar to the input mention before being re-ranked by a more fine-grained encoder. On the other hand, the list of candidates is not entirely rejected if the entry does not correspond to any entity in the database (NIL prediction). To overcome this issue, Zhang et al. (2021) applies a Q&A approach, returning the entities likely to be in the document before classifying whether a textual segment mentions a candidate. The NIL classification is only partially solved since they cannot add the description of a new entity to predict it later. In addition, Blink and EntQA use three BERT encoders with separate weights which makes the system quite cumbersome.

2.3 KBP Evaluation

While there are numerous benchmarks and metrics for the evaluation of subtasks ("F1-score" for REN and relation extraction, "Hit@k" for linking, etc), few solutions exist for end-to-end systems that build knowledge bases from text. The TAC KBP workshops evaluate a system by computing the accuracy on 1-hop queries, *"What is Frodo carrying?"*, and 2-hop queries, *"Who created what Frodo is carrying?"*, the number of hops representing the number of relations which separate the subject entity from the object entity. The cost of manually evaluating systems that return a large number of responses (Ellis et al., 2015) compels evaluators to focus on a small number of queries and thus they do not evaluate the entire database. Min et al. (2018) makes automatic evaluation feasible by measuring the alignment of triplets *(subject, relation, object)* between the reference and the output. An output entity is linked to a reference one if the produced entity shares more than 50% of the mentions with the reference entity. This alignment raises the questions of the situation where the system would only extract a small number of mentions and also the arbitrary choice of the threshold at 50%. KnowledgeNet (Mesquita et al., 2019) measures the $F1$ score on the extraction of annotated triples in sentences and the linking of the subject and object entity pair to their Wikidata page. Since each sentence in the dataset annotates only one pair of entity and one relation, it is not possible to properly evaluate the accuracy since results could be mistakenly considered as false positives. Like Min et al. (2018), the evaluation is done at the textual level and discards the construction of a base. These incomplete evaluation methods highlight the need for a protocol that evaluates the performance of an entire KBP system.

3 MODELING AND EVALUATION

3.1 Modeling a Knowledge Base

A KB is composed of elements (entities, attributes and relations between them) relying on a defined ontology. It can thus be modeled by a graph in which the nodes are the various elements and the edges express the existence of a relation between these elements. We define a KB as follows:

Definition 3.1 (Knowledge Base)**.** *A Knowledge Base is a data structure that can be modeled by a graph* $G = (V, E, \Phi, \Psi)$ *where* V *is the set of vertices in the graph,* E *the set of edges between two nodes of* V*,* $\Phi : V \rightarrow \mathcal{A}$ *is a function which for any vertex* v_i *of* V *assigns a set of attributes* $A_i \in \mathcal{A}$ *denoted by tuples (type, value) and* $\Psi : E \rightarrow \mathcal{E}$ *a function that associates to each edge* $e_i \in E$ *an edge type* $E_i \in \mathcal{E}$*, with* $\mathcal{A}$ *and* $\mathcal{E}$ *referring to the set of attributes and the set of edge types respectively.*

Populating a KB with textual content consists therefore in adding elements extracted from texts according to an ontology. To link information related to the same entity found in several texts, the entities must have a unique identifier (URI). This allows to obtain for a set of k texts, a reference KB, $G_k = (V_k, E_k, \Phi_k, \Psi_k)$ and to measure the proportion of information correctly extracted by a system building a base $G'_k = (V'_k, E'_k, \Phi'_k, \Psi'_k)$.

Example of workflow for the KBP task. A KBP system is built around components or modules that form the processing workflow to solve the KBP task. The first component is in charge of recognizing named entities (NER) and other elements of interest in the text (attributes, unnamed entities) while assigning them a type using the document. For instance with the sentence "*Joe Biden, the U.S. President, went viral on Trump*", the step yields [*(Joe Biden, Per), (the U.S. President, Per), (President, Role), (U.S., Nationality), (Trump, Per)*]. The second processing block groups the textual elements that co-reference. The mentions from the NER belonging to a cluster composed of mentions of the same type are kept in coreference and the remaining mentions are considered as different entities. Using the previous example, we get two clusters [*(1, PER, Joe Biden, the U.S President), (2, PER, Trump)*]. It is then possible to identify, by a third module, the relations linking the elements [*(1, Is against, 2), (1, Role, President), (1, Nationality, U.S.)*]. These relations, in addition to being part of the information to be extracted, constitute a support for the last step, the entity resolution. Each entity in the text, when it is possible, is associated with an entity in the database. "*Trump*" must be linked to the entity "*Donald Trump*" and not to "*Fred Trump*". Finally, all the information extracted from the text enriches the information in the database by completing those of the entities already known or by adding new entities.

3.2 Evaluating KBP Systems

In order to evaluate the systems designed for the KBP task, we present below a process to measure the performance of the methods which extract and aggregate information either to an existing base (warm start) or to an initially empty base (cold start).

Entities are defined by attributes and relations which link them to other entities in the database. When comparing a reference entity to a built entity, it is necessary to verify that both the attributes and the relation match.

Definition 3.2 (Similarity of attributes and relations)**.** *We use the following similarity definitions for attributes and relations:*

- ***Attribute similarity****: we consider that 2 attributes match if they have the same type, value and inference text (in which the attribute appears). Although including the inference text creates a multiplication of information, it verifies that the system correctly extracts the information each time it is mentioned.*
- ***Relation similarity****: we consider that 2 relations are similar if they involve the same predicate (type of relation), the same inference text and that all the mentions of the object entity of the constructed relation are included in the mentions of the object entity of the reference base.*

To check if an entity has been correctly extracted, we compare the extracted attributes and relations with those possessed by the reference entity. In order to measure the proximity, we adapt the precision, recall and F1-score based on Definition 3.2 as follows:

$$
\begin{aligned}
P_{v_i,v_j,k} &= \frac{\alpha|TP_{rel}| + \beta|TP_{att}|}{\alpha(|TP_{rel}| + |FP_{rel}|) + \beta(|TP_{att}| + |FP_{att}|)} \\
R_{v_i,v_j,k} &= \frac{\alpha|TP_{rel}| + \beta|TP_{att}|}{\alpha(|TP_{rel}| + |FN_{rel}|) + \beta(|TP_{att}| + |FN_{att}|)} \\
F1_{v_i,v_j,k} &= 2\frac{P_{v_i,v_j,k} \times R_{v_i,v_j,k}}{P_{v_i,v_j,k} + R_{v_i,v_j,k}}
\end{aligned}
\tag{1}
$$

With TP, FP and FN for true positives, false positives and false negatives respectively, *rel* for relation and *att* for attribute v_i and v_j vertices belonging to the reference base G_k and the built base G'_k, and $0 \leq \alpha, \beta \leq 1$

weights such as $\alpha+\beta=1$, which allow to give a different importance to attributes and relations. Other weights could be added to differentiate along attribute or relation types.

Entity alignment. We align each entity of the reference KB with an entity of the output base using the $F1$ score defined above and the Hungarian algorithm (Kuhn, 1955). The alignment is possible only for pairs with a non-zero similarity score. G_k and G'_k entities without match are respectively considered as false negatives and false positives. In the warm-start scenario, entities that are initially present remain aligned between G_k and G'_k, their F1-score only takes into account the new information. This matching phase leads to the construction of Ω_k a set of pairs (v_{i,G_k}, v_{j,G'_k}).

Global quality scores. The comparison between the constructed and the reference bases after proceeding k texts is done by aggregating the similarity scores of the previously formed pairs of entities. Two F1-scores, one $F1_{micro}$ and one $F1_{macro}$ measuring the proportion of correctly extracted information, are computed:

$$
\begin{aligned}
P_{micro,k} &= \frac{\sum_{(v_i,v_j)\in\Omega_k} \alpha|e_{v_j}=e_{v_i}|+\beta|a_{v_j}=a_{v_i}|}{\alpha|E_{G'_k}|+\beta|A_{G'_k}|} \\
R_{micro,k} &= \frac{\sum_{(v_i,v_j)\in\Omega_k} \alpha|e_{v_j}=e_{v_i}|+\beta|a_{v_j}=a_{v_i}|}{\alpha|E_{G_k}|+\beta*|A_{G_k}|} \\
F1_{micro,k} &= 2\frac{P_{micro,k}\times R_{micro,k}}{P_{micro,k}+R_{micro,k}} \\
F1_{macro,k} &= \frac{\sum_{(v_i,v_j)\in\Omega_k} F1_{v_i,v_j,k}}{|\Omega_k|+|FN|+|FP|}
\end{aligned}
\tag{2}
$$

With e_{v_i}, e_{v_j} edges of node i and j and E_{G_k}, A_{G_k}, respectively the set of edges and attributes in the reference base. The $F1_{macro}$ is an average of the similarity scores of the aligned entities and does not take into account the difference of distribution (number and type of relations or attributes) that could exist between the entities, unlike the $F1_{micro}$. The latter is a weighted F1 calculated according to the identical elements between the aligned entities.

Benefits of the evaluation process. An issue when considering an end-to-end system, which is composed of several modules, is that an error caused by a module can be re-used by another and added to the database. For example, a person mentioned in a text can be linked to the wrong person in the database and thus can lead to an erroneous assignment of a relation or attribute. The proposed protocol measures at different text intervals the distance to the baseline, showing the resilience of a system to errors that can be made. The evaluation can be conducted both in a warm-start and a cold-start scenario. The impact of a single module on the whole processing chain is measured using the ground truth results on the rest of the workflow. The choice to make the F1-score measurement more flexible, by replacing the exact matching of entities by the proportion of identical information in a pair, brings a better representativeness of the systems' performances.

4 MODELS FOR KBP

This section presents ELROND, a baseline system for the KBP task and MERIT, an entity linking task module which improves the baseline.

4.1 ELROND, an End-To-End System for KBP

We introduce ELROND (Entity Linking and Relation extraction On New textual Documents). ELROND is an implementation that follows the KB enrichment process explained in Section 3.1. The main components, with their interactions, are illustrated in Figure 1. This system is used as a baseline and shows the interest of the evaluation method detailed in Section 3.2. For each module we implement a recent proposal found in literature which exhibits good results.

Named Entity Recognition. The NER block consists of a pre-trained and fine-tuned RoBERTa model (Liu et al., 2019). The choice of RoBERTa was motivated by its current performance on various benchmarks (*SWAG*[1], *GLUE*[2]) for named entity recognition.

Resolving co-references. For the co-reference task, we use in parallel of the NER model, the pre-trained model *Word-level Coreference Resolution* (Dobrovolskii, 2021). This model creates, with the help of RoBERTa, groups of words in co-reference (named and unnamed entities, pronouns, etc). The representation of a token is given by weighting the vectors of its sub-tokens produced by RoBERTa. The weights are obtained by applying a softmax function to the projection of the vectors through an attention matrix. Finally, the model predicts a co-reference when the

[1] https://paperswithcode.com/sota/common-sense-reasoning-on-swag

[2] https://gluebenchmark.com/

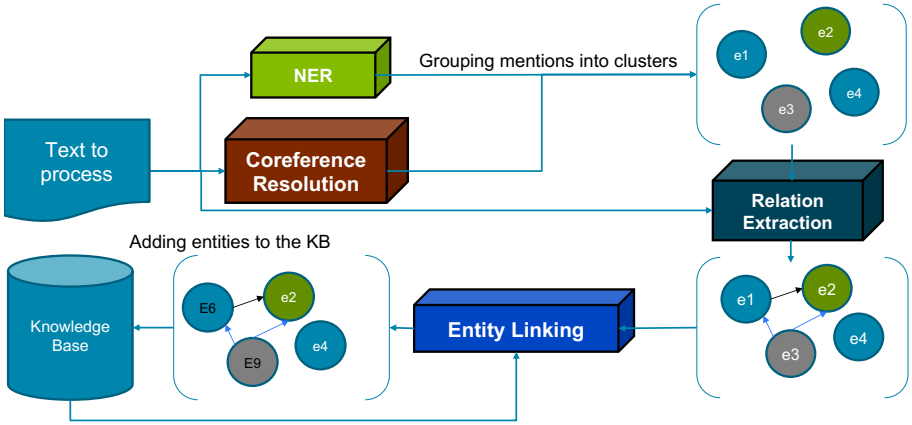


Figure 1: Diagram of the proposed processing chain for ELROND.

sum of a bilinear projection between two tokens and the output of a neural network taking the two tokens as input is positive. We choose to integrate *Word-level Coreference Resolution* to ELROND because of its performance on this task.

Relation Extraction. Relations are extracted using the ATLOP model (Zhou et al., 2021) which represents each entity by applying a pooling function on the mention vectors obtained by a PLM (Pre-trained Language Model). For each pair of entities, an attention coefficient is obtained before using it in a bilinear function to compute the plausibility of a relation type. If the score is greater than the Null type, the relation is considered as existing. Relationships that are impossible due to the type of entities are filtered in post-processing.

Entity Linking. The last step applies a search by mention and a selection by popularity. For each entity in the text, the solution returns the entities in the database, of the same type and sharing at least one mention with the entity in the text. In case the mentions do not return any results, an extended search is performed with the acronyms of these mentions. If no element is returned, a the textual entity is added to the database. If several entities of the database correspond to mentions of the textual cluster, a selection by popularity, similar to Al-Badrashiny et al. (2017) is applied. The entity with the most occurrences, considering all mentions, is selected.

4.2 MERIT: Model for Entity Resolution In Text

Candidate Retriever. The popularity-based linking approach described previously that serves as baseline is not usable for dealing with new mentions and is prone to errors during ambiguity resolution, since it favors popular entities. The proposed method, MERIT, addresses these shortcomings by relying on the context of the documents, drawing inspiration from previous approaches such as Blink (Wu et al., 2020) or EntQA (Zhang et al., 2021). Like these models and as illustrated in figure 2, we propose as a first component an encoder that projects a portion of a text targeting an entity to a representation space allowing similarity comparisons. The retriever takes as input a query text in which the target entity mentions are enclosed in tags. This retriver differs from EntQA and BLINK on several points. First, the text samples are expanded to a size of 256 tokens to include a broader context for a better discrimination. Secondly and as illustrated by Soares et al. (2019) all mentions of the entity of interest in the text sample are wrapped with special tags (*[Ent]* and *[/Ent]*) to improve the quality of representations. The distance between the vector representations of the text samples is measured using cosine similarity instead of a scalar product. In addition to obtaining better performance in our case, Luo et al. (2018) show that this provides a more stable learning. Lastly, a simple encoder replaces the dual-encoder since the search base is no longer composed of descriptions, but with textual portions of the same style as those given in the query. The BERT encoder is replaced by the ALBERT architecture, which is lighter and more efficient for semantic similarity tasks (according to the STS benchmark[3]).

Classifier. The most similar text samples retrieved are then concatenated with the query one. The set of 512 tokens is given to an ALBERT model with a lin-

[3]https://paperswithcode.com/sota/semantic-textual-similarity-on-sts-benchmark

ear layer for classification at the end. If none of the samples are classified as similar, a new entity is created in the base with the query sample as the first support text. In the case where several samples are classified, the one with the higher classification score is selected. The model then returns as output the database entity that corresponds to the request entity (if a match has been established).

5 EXPERIMENTS

5.1 Implementation details

All the proposed approaches are implemented in Python and use the *Pytorch* library[4]. The NER model is trained using the Flair (Akbik et al., 2019) framework , the encoders used by MERIT are initialized from the library *Hugging Face* [5]. The MERIT retriever module training uses the contrastive loss (Khosla et al., 2020) which leads to a better clustering of identical elements in a representational space. We use the hard-negative technique (Gillick et al., 2019) which consists in submitting negative samples considered as ambiguous by the model during training. This process brings a better selection of parameters and makes the model more robust in its predictions. The classification module is trained by the binary cross-entropy function. We use the index structure *Annoy*[6] to return elements according to their cosine similarity.

5.2 Datasets

For a thorough measurement of systems, the completeness of dataset annotation on all dimensions of the information to be extracted is necessary. We decide therefore to use DWIE (Zaporojets et al., 2021), the only free dataset complying with this constraint. The dataset is composed of 800 press articles in English. In the 700 training texts and 100 test texts that constitute DWIE, entities are annotated according to a multi-level ontology for a total of about 170 classes and relations. For the KBP task, types and aliases are used as attributes. The 700 training texts constitute the KB to be completed in Warm-start.

We also used AIDA (Hoffart et al., 2011) to compare the linking approch using mentions popularity to MERIT. AIDA consists of 1393 news articles for which entities and their Wikipedia page names are listed when available. For DWIE, entities that are not linked to Wikipedia only appear in a single text, so we can consider them as unique entities. This is not the case for AIDA. Entities in AIDA that do not have an identified link are thus discarded from the linking task.

To estimate the complexity of the linking task on a dataset, we introduce an ambiguity score, χ, computed as the average for each textual mention, of the inverse of the number of entities named by it:

$$\chi(dataset) = \frac{\sum_{mentions} |E(mention)|^{-1}}{|mentions|} \quad (3)$$

With $E(mention)$ the set of entities sharing this mention. We obtain a score of $\chi(DWIE) = 0.889$ and $\chi(AIDA) = 0.794$.The entity linking task is therefore more complex for the dataset AIDA.

Since the existence of certain items in a database, and thus the order in which the texts allowing the extraction of these items are submitted, may or may not benefit the operation of the systems, the performance is measured and averaged over 10 different orderings of the test set. The data and scripts to compare these results are available on a git[7] repository.

5.3 Results and Discussion

ELROND**'s performance.** The graphs in Figure 3 show the score of ELROND for the KBP task on the 100 test texts (average taken over 10 runs). For the warm-start scenario, the original KB is the information contained in the 700 training texts. We observe that the initial distance between the KBs is greater in warm-start due to the difficulty to link the information of the texts with those already possessed. In both cases (although more contrasted in warm-start), the performance declines over the texts, which attests of an accumulation of errors during the process. The micro F1-score is higher than the macro F1-score in both cases and seems more stable over the end of the test set. This is explained by the fact that popular entities (countries and cities for example) which are mentioned more often in the texts have on the one hand more weight in the final database, and on the other hand are easier to recognize. This characteristic will therefore tend to increase the $F1_{micro}$ compared to the $F1_{macro}$ which smoothes the difference in distribution between entities.

Comparison of linking approaches. For entity resolution, we use two types of metrics. The $Hit@k$, that computes the frequency with which the queried entity is found among the first k entities. This metric is only

[4]https://pytorch.org/

[5]https://huggingface.co/

[6]https://github.com/spotify/annoy

[7][https://github.com/Todaime/KBP]

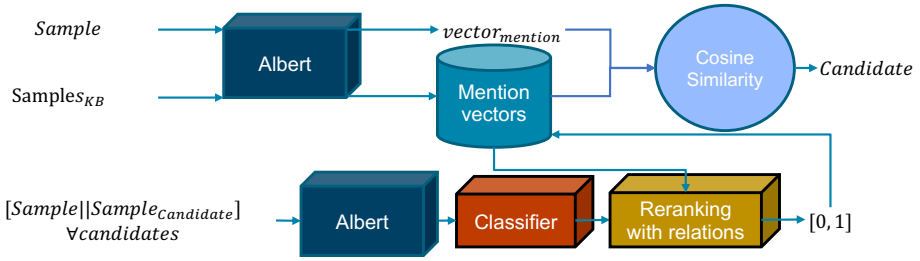
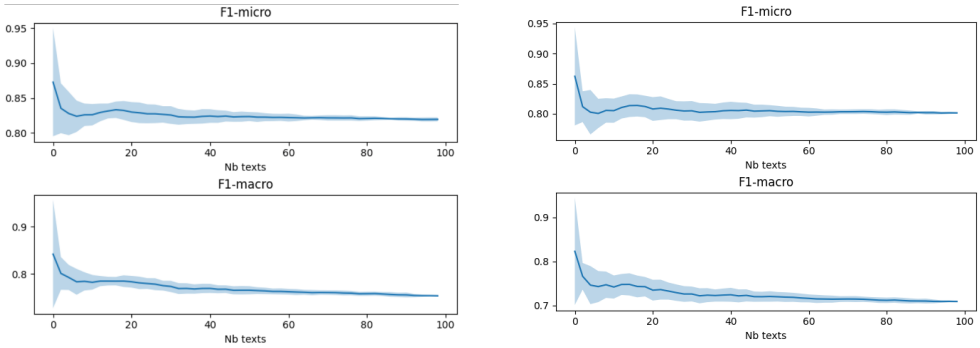


Figure 2: Diagram of MERIT.



(a) Cold-start (b) Warm-start

Figure 3: ELROND performance for the KBP task on DWIE texts.

measured on query entities that are in the database. The second metric is *accuracy*: a result is valid if the entity is in the KB and is returned by the model or if this entity is NIL and the model does not return a result. We add for comparison a linking solution that randomly select an entity in cases of ambiguity. For DWIE, we measure the performance with and without filtering on the type of entities as well as taking into account the relations that exist in the KB. Due to the lack of annotation on AIDA, the performances of the approaches do not those elements. To reduce the prediction of false positives, MERIT retains only the first 10 candidates obtained by similarity search.

All the results are recorded in tables 1 and 2. For similarity retrieval and for both datasets, MERIT gives better results than the mentions popularity approach and shows a benefit when used to propose results during a semi-automatic process. The trend is more nuanced on DWIE when dealing with NIL entities. This is explained by the fact that MERIT applies a classification on more candidates and is therefore more prone to predict false positives, while the mention popularity approach filters out identical mentions and is limited to 1 or 2 candidates. AIDA contains more ambiguity in its texts but with distinct writing styles and contexts between entities sharing mentions. This may explain why MERIT has better results than the mentions popularity module. We observe that re-ranking the returned entities according to relationship consistency slightly improves the results. We also studied a combination of the two approaches. This fusion use MERIT with a classification on the most similar candidate and selects the search results by mention popularity in case of negative classification. The results show in this case study a slight interest for this fusion when applied to DWIE, but are finally less interesting on AIDA. The drop on AIDA is explained by the influence of the search by popularity of mentions.

Finally, we compared for the KBP task ELROND, ELROND with the Fusion entity linking solution and the model proposed in the DWIE paper. Since no linking method is given for the latter, we used the mention popularity method. The results presented in table 3 show the interest of the proposed approaches which improve the results by up to 2.2% in the Warm-start scenario, and the complementarity of MERIT with the mention approach.

6 CONCLUSIONS

In this paper, we have formalized and presented an automatic, complete and scalable evaluation method for the KBP task from texts. It allows to compare and select methods in warm-start and cold-start scenarios. This protocol has been used to measure the performance of ELROND, a system that serves as a first basis for future improvements. We were able to improve

	DWIE			
Model	Hit@1	Hit@10	Hit@50	Accuracy
Unfiltered				
Random Linking	84.7	92.5	92.5	91.9
Mentions	89.8	93.6	93.6	92.2
MERIT	91.8	95.4	95.8	93.0
Fusion	92.1	95.2	95.8	92.6
Filtered				
Random Linking	88.5	93.1	93.1	93.7
Mentions	91.5	93.6	93.6	94.6
MERIT	94.2	96.5	**96.7**	93.9
MERIT+Mentions	94.3	**96.6**	**96.7**	94.6
Filter + relations				
MERIT	**95.0**	96.5	**96.7**	94.1
MERIT+Mentions	94.4	96.5	**96.7**	**94.8**

Table 1: Performance of the linking approaches on the DWIE dataset.

	AIDA			
Model	Hit@1	Hit@10	Hit@50	Accuracy
Random Linking	77.8	92.0	92.5	82.8
Mentions	77.5	92.6	92.6	82.8
MERIT	**94.6**	**98.2**	**98.6**	**92.1**
Fusion	94.2	97.6	97.9	90.1

Table 2: Performance of the linking approaches on the AIDA dataset.

the entity linking module of the ELROND baseline by proposing MERIT, an entity resolution model based on textual encoders and capable of integrating NIL entities during inference. Future work may study the contribution of a larger dataset for training a supervised linking model, extend the elements of interest to unnamed entities, to study linking approaches that would only have a structured KB and the role that ontology can play in KBP systems. We are also working on the production of a French dataset to train and evaluate KBP systems. We plan to extend the presented models to the French language.

REFERENCES

Akbik, A., Bergmann, T., Blythe, D., Rasul, K., Schweter, S., and Vollgraf, R. (2019). Flair: An easy-to-use framework for state-of-the-art nlp. In *Proceedings of the 2019 conference of the North American chapter of the association for computational linguistics (demonstrations)*, pages 54–59.

Al-Badrashiny, M., Bolton, J., Chaganty, A. T., Clark, K., Harman, C., Huang, L., Lamm, M., Lei, J., Lu, D., Pan, X., et al. (2017). Tinkerbell: Cross-lingual cold-start knowledge base construction. In *TAC*.

Ammar, W., Groeneveld, D., Bhagavatula, C., Beltagy, I., Crawford, M., Downey, D., Dunkelberger, J., Elgohary, A., Feldman, S., Ha, V., et al. (2018). Construction of the literature graph in semantic scholar. In *Proceedings of NAACL-HLT*, pages 84–91.

Dobrovolskii, V. (2021). Word-level coreference resolution. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 7670–7675.

Ellis, J., Getman, J., Fore, D., Kuster, N., Song, Z., Bies, A., and Strassel, S. M. (2015). Overview of linguistic resources for the tac kbp 2015 evaluations: Methodologies and results. In *TAC*.

Ghosal, T., Al-Khatib, K., Hou, Y., de Waard, A., and Freitag, D. (2022). Report on the 1st workshop on argumentation knowledge graphs (argkg 2021) at akbc 2021. In *ACM SIGIR Forum*, volume 55, pages 1–12. ACM New York, NY, USA.

Gillick, D., Kulkarni, S., Lansing, L., Presta, A., Baldridge, J., Ie, E., and Garcia-Olano, D. (2019). Learning dense representations for entity retrieval. In *Proceedings of the 23rd Conference on Computational Natural Language Learning (CoNLL)*, pages 528–537.

Hoffart, J., Yosef, M. A., Bordino, I., Fürstenau, H., Pinkal, M., Spaniol, M., Taneva, B., Thater, S., and Weikum, G. (2011). Robust disambiguation of named entities in text. In *Proceedings of the 2011 conference on empirical methods in natural language processing*, pages 782–792.

Huang, Z., Xu, W., and Yu, K. (2015). Bidirectional lstm-crf models for sequence tagging. *arXiv preprint arXiv:1508.01991*.

	Cold-start		Warm-start	
Model	F1-Micro	F1-Macro	F1-Micro	F1-Macro
ELROND+FUSION	**83.5**	**76.3**	**82.0**	**72.1**
ELROND	83.4	76.1	81.4	**72.1**
DWIE	82.8	75.6	80.3	69.9

Table 3: Final model scores for the KBP task on the DWIE dataset.

Kenton, J. D. M.-W. C. and Toutanova, L. K. (2019). Bert: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of NAACL-HLT*, pages 4171–4186.

Khosla, P., Teterwak, P., Wang, C., Sarna, A., Tian, Y., Isola, P., Maschinot, A., Liu, C., and Krishnan, D. (2020). Supervised contrastive learning. *Advances in Neural Information Processing Systems*, 33:18661–18673.

Kuhn, H. W. (1955). The hungarian method for the assignment problem. *Naval research logistics quarterly*, 2(1-2):83–97.

Lin, X., Li, H., Xin, H., Li, Z., and Chen, L. (2020). Kbpearl: a knowledge base population system supported by joint entity and relation linking. *Proceedings of the VLDB Endowment*, 13(7):1035–1049.

Liu, Y., Ott, M., Goyal, N., Du, J., Joshi, M., Chen, D., Levy, O., Lewis, M., Zettlemoyer, L., and Stoyanov, V. (2019). Roberta: A robustly optimized BERT pretraining approach. *CoRR*.

Luo, C., Zhan, J., Xue, X., Wang, L., Ren, R., and Yang, Q. (2018). Cosine normalization: Using cosine similarity instead of dot product in neural networks. In *International Conference on Artificial Neural Networks*, pages 382–391. Springer.

Mesquita, F., Cannaviccio, M., Schmidek, J., Mirza, P., and Barbosa, D. (2019). Knowledgenet: A benchmark dataset for knowledge base population. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 749–758.

Min, B., Freedman, M., Bock, R., and Weischedel, R. (2018). When ace met kbp: End-to-end evaluation of knowledge base population with component-level annotation. In *Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC 2018)*.

Shue, L.-Y., Chen, C.-W., and Shiue, W. (2009). The development of an ontology-based expert system for corporate financial rating. *Expert Systems with Applications*, 36(2):2130–2142.

Soares, L. B., Fitzgerald, N., Ling, J., and Kwiatkowski, T. (2019). Matching the blanks: Distributional similarity for relation learning. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 2895–2905.

Wu, L., Petroni, F., Josifoski, M., Riedel, S., and Zettlemoyer, L. (2020). Scalable zero-shot entity linking with dense entity retrieval. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 6397–6407.

Yao, Y., Ye, D., Li, P., Han, X., Lin, Y., Liu, Z., Liu, Z., Huang, L., Zhou, J., and Sun, M. (2019). Docred: A large-scale document-level relation extraction dataset. In *ACL (1)*.

Zaporojets, K., Deleu, J., Develder, C., and Demeester, T. (2021). Dwie: An entity-centric dataset for multi-task document-level information extraction. *Information Processing & Management*, 58(4):102563.

Zhang, W., Hua, W., and Stratos, K. (2021). Entqa: Entity linking as question answering. In *International Conference on Learning Representations*.

Zhou, W., Huang, K., Ma, T., and Huang, J. (2021). Document-level relation extraction with adaptive thresholding and localized context pooling. In *Proceedings of the AAAI conference on artificial intelligence*, volume 35, pages 14612–14620.