information

Article

Computationally Efficient Context-Free Named Entity Disambiguation with Wikipedia

Michael Angelos Simos [1,*] and Christos Makris [1,2,*]

[1] Department of Computer Engineering and Informatics, University of Patras, 26504 Patras, Greece
[2] School of Science and Technology, Studies in Informatics, Hellenic Open University, Member of the Collaborating Teaching Staff, Parodos Aristotelous 18, 26335 Patra, Greece
* Correspondence: asimos@ceid.upatras.gr (M.A.S.); makri@ceid.upatras.gr (C.M.); Tel.: +30-2610-996-968 (C.M.)

Abstract: The induction of the semantics of unstructured text corpora is a crucial task for modern natural language processing and artificial intelligence applications. The Named Entity Disambiguation task comprises the extraction of Named Entities and their linking to an appropriate representation from a concept ontology based on the available information. This work introduces novel methodologies, leveraging domain knowledge extraction from Wikipedia in a simple yet highly effective approach. In addition, we introduce a fuzzy logic model with a strong focus on computational efficiency. We also present a new measure, decisive in both methods for the entity linking selection and the quantification of the confidence of the produced entity links, namely the *relative commonness* measure. The experimental results of our approach on established datasets revealed state-of-the-art accuracy and run-time performance in the domain of fast, context-free Wikification, by relying on an offline pre-processing stage on the corpus of Wikipedia. The methods introduced can be leveraged as stand-alone NED methodologies, propitious for applications on mobile devices, or in the context of vastly reducing the complexity of deep neural network approaches as a first context-free layer.

Keywords: named entity disambiguation; text annotation; context-free Wikification; word sense disambiguation; ontologies; Wikification; fast Wikification; artificial intelligence; machine learning

check for updates

Citation: Simos, M.A.; Makris, C. Computationally Efficient Context-Free Named Entity Disambiguation with Wikipedia. *Information* **2022**, *13*, 367. https://doi.org/10.3390/info13080367

Academic Editor: Sarantos Kapidakis

Received: 29 May 2022
Accepted: 29 July 2022
Published: 2 August 2022

1. Introduction

The natural languages' immanent peculiarities introduce several challenges in the semantic interpretation of unstructured text corpora. Specifically, the linguistic phenomena of homonymy and polysemy entail the coexistence of diverse potential perceptions for a word or phrase in different occurrences and contextual information backgrounds. The research area of Artificial Intelligence (AI) and Natural Language Processing (NLP) engaged with this class of predicaments is known as Named Entity Disambiguation (NED). The focus of this domain is the semantic resolution, convergence, and assignment, of a textual mention or text unit, to a corresponding entity concept from an ontology or knowledge base.

The research area of NED domain is gaining popularity at the limelight of recent works in the semantic search, web search, information retrieval, and data mining, as deeper knowledge acquisition is essential for attaining more expressive and, hence, more accurate results, in contrast with the widely adopted semantically superficial approaches exhibiting knowledge-acquisition impediments. Subsequently, the NED task is crucial in knowledge extraction processes for research and commercial purposes in the broader AI, information, and Internet industries.

Wikipedia is one of the largest online general knowledge sources of encyclopedic structure. It consists of millions of articles, which are developed and maintained by the convergence of abundant points of view from a large online community of active contributors, editors, and administrators. Therefore, each such article tends to summarize a consensus of a semantic concept, hence, the entire structure of Wikipedia can be leveraged and interpreted

as an ontology or knowledge base. The affluent contextual and link structure within the corpus of Wikipedia articles is being fruitfully exploited by several similar works, for high-performing NED applications, in a task that is commonly known as Wikification in the relevant literature [1–9].

The maintenance of ontologies and knowledge sources in general has been among the key adversities on the NED task, as concepts tend to change over time while new conceptual areas emerge. The integration of Wikipedia as a knowledge source in the task, led to considerable advances in the knowledge acquisition challenge and substantial progression at the antagonistic knowledge source resolution, as the article knowledge-convergence process is leveraged for deriving a set of widely accepted textual descriptions for a vast set of concepts in the form of encyclopedic articles.

Research works on several AI applications and fields have adopted deep neural network architectures with noteworthy success in the domains of computer vision [10], image analysis and processing [11], audio and speech recognition [12–14], fraud detection [15], healthcare [16–18], autonomous driving [19], natural language processing [18,20,21] etc. Essentially, deep learning architectures have near-human accuracy.

Deep neural networks are being exploited as black boxes for the purpose of mapping input data to classification or approximation outputs, and the mentality of interpretation of the internal workings is often absent in the data science lifecycle process due to the inherent challenges and elevated complexity of the task. This interpretability problem of deep learning architectures [22,23] is outlined as one of the main adversities, and in several use-cases can be an inhibitory factor for adoption despite the impressive predictive accuracy for business-critical applications and mission-critical processes [23]. The evolution of AI integration on such applications preconditions faith and trust for the machine output, which is essentially dependent on interpretability in terms of a human's understandable rationalization for the machine output, hence, intuition regarding the internals of deep learning models. Interpretability is a trending field in the machine learning domain, as the extraction of knowledge regarding relationships that are accommodated in the data or the model as visualizations or mathematical equations, for instance, can derive insights that could drive actions and research.

Deep neural network methodologies come at a considerable, computational cost for attaining state-of-the-art accuracy on several AI tasks and applications. According to Thompson et al. [24], the progress on AI tasks including named entity recognition, which have been thoroughly reviewed, has been identified as strongly dependent on vast computational resources. The projections indicate several sustainability challenges on the progress of deep neural network approaches, in terms of technical, environmental, and economical aspects. The authors call out the necessity to explore more efficient approaches, for supporting sustainable progress on AI fields, which would require the exploration of other machine learning approaches or radical changes on deep learning. As also noted in [24], deep learning computational complexity is inherent to its design, although the lower theoretical bounds do not seem to follow the computational requirements scaling trend, thereby implying that optimizations on that regard may be as well feasible.

Knowledge attainment in the form of the intricacy to obtain sense-annotated text corpora is among the main strains of the NED task, largely known as the knowledge-acquisition bottleneck [25–27]. Successful deep learning methods are data-hungry, requiring extensive training datasets [25,26]. To that end, the current work is focused on a semantic ontology graph representation of Wikipedia, leveraging the rich information of contextual content and hyperlinks present at its corpus. Another noteworthy adversity for deep neural network implementations is believed to reside on the catastrophic forgetting problem [28], posing limitations on model generalization. In addition, the vast input dimensionality of the problem introduces representation and computational efficiency challenges, as summarized in [29]. In that regard, some recent research works outline the impediments of production grade big data applications based on deep neural networks, for real-world applications [2].

As the focus is generally aimed towards prediction precision rather than run-time efficiency, there is an increasing demand for performance-oriented yet accurate methodologies.

Considering the ICT sector contribution estimates range from 2% according to [30] to 4%, including the supply chain pathways according to the more recent data from [31] and the global challenges in the post-COVID era, in this work, we are exploring some differential perspectives to deep learning approaches for the NED task. Specifically, we explore some performance-oriented approaches for the NED to the Wikipedia Entities problem, as an alternative to deep learning architectures, aiming for high precision with computational efficiency, with a high degree of scalability in mind. Since deep neural network based methods have been dominating research fields and intricate application domains, the big-data industry retains a focus on high performance applications, rendering our method a good candidate for adoption on bigdata.

This manuscript is organized in the following sections: An overview of the related work along with the relevant work background is covered in Section 2. A presentation of our methods and their implementation specifics are detailed in Section 3. An analysis and discussion of our experimental evaluation approach and results is presented in Section 4, in conjunction with some future work items. A final summary of our results is conferred along with our conclusions in Section 5.

2. Brief Related-Work Background

The methodologies for NED may be classified on several aspects. Disambiguation can be aiming, for instance, to an ontology level linking or the broader domain identification. The scope of disambiguation may be focused on a single concept or anchor within a given text or even any candidate anchor that can be a linked to an entity. From a machine learning perspective, according to Navigli [25], NED can be supervised, hence relying on semantically annotated corpora or knowledge bases, or unsupervised, namely reliant on unstructured text datasets that can be leveraged for semantic inference. In addition, depending on the kind of sources of knowledge employed, we may distinguish knowledge-rich methods, relying on ontologies, thesauri, or other lexical resources and knowledge-poor methods relying on unstructured text corpora. The focus of this work is an approach for the NED problem with Wikipedia, a topic commonly referred to as Wikification. Due to the complexity of the task, a literature overview is best fit for briefly covering the related works background and the motivation of the current work.

2.1. Named Entity Disambiguation with Wikipedia Entities Methodologies

The concept of Named Entity Disambiguation [32] was expanded to Wikipedia Entities, initially by [1,6]. These first works have set the basis for further research, while proposing some first approaches to the problem leveraging the commonness, namely the prior probability of linking a mention to a specific entity as a feature for the disambiguation phase. The introduction of relatedness, in the form of a measure of overlapping inbound inter-wiki links between Wikipedia entities followed by [3,7], quantifyies the semantic relevance among Wikipedia entities, for building a coherence based ranking system for ambiguous text anchors. Scoring based on the semantic detachment of global and local context, based on consensus attainment of both, in a ranking formalization, led to further improvements in [5]. A framework combining previous work, aiming at short-text-input Wikification with computational efficiency was proposed in [8], based on a voting structure and a ranking and filtering phase. In [33], a graph representation is utilized modeling the main features of previous works, along with contextual relatedness, utilizing a graph-search approach for deriving disambiguation. A similar graph representation is used in [34], using a PageRank-based methodology. An exhaustive technique and an iterative method are presented by [4], employing Wikipedia graph features. In [9], the TAGME methodology is revisited, with the exploration of several variants and a contribution of a set of methods. Finally, in [2], the authors present a Wikification suite focusing on run-time-efficiency oriented yet efficacious approaches, for big-data adoption. According to our research, the

methods described in [2] attain competitive run-time performance, improving previous state-of-the-art approaches for fast Wikification such as [9], in terms of run time, by at least one order of magnitude.

2.2. Recent Compute Intensive Approaches

The authors of [35,36] presented some of the first deep learning approaches, using a vector space representation. In [35], mainly a convolutional neural network and a tensor network model are employed for the disambiguation and pruning phases. In [35], a skip-gram model extension based on a KB graph and a vector-proximity measure are being used in the process. An ensemble-learning approach is described by [37], involving statistics and a graph representation, entity embeddings, and processing on variable context scopes. The work in [38] has a differentiated approach relying on the unsupervised extraction of semantic relations and an optimization process for the final-mention selection, exhibiting promising quality-performance experimental results. Another knowledge-graph-based methodology is employed for the problem by [39], involving entity embeddings and transformations on the knowledge-graph density, through coreference statistics inference from unstructured text corpora. The authors of [40] are shifting the focus to denser context windows, via a sequence-decision modeling of the task, by leveraging an exclusion process for non-consistent candidate mentions before employing a reinforcement learning neural network in a wider context. Another noteworthy work is presented in [41], with a BERT-based model for attaining an accuracy and speed balance, depicting, similarly to [2], the importance of run-time performance for industry adoption, yet requiring substantial resources.

3. Materials and Methods

The Wikification process on several similar approaches and the current work involves four main phases that will be described in this section:

1. Extraction, transformation, and loading phase of our underlying knowledge base.
2. Mention parsing and identification of candidate mentions within the unstructured text input.
3. Mention disambiguation/entity linking for selecting mention annotations.
4. Mention annotation scoring and pruning based on confidence evaluation.

The terminology and notation applied throughout this section follows the established literature norms. More specifically:

- The notion p will be used for Wikipedia articles, i.e., entities.
- A *mention* will refer to a hyperlink to a p.
- A mention to a p will be referred to as a. Consequently, sequences of such mentions may be using indexing for reference, starting from a_1 and ranging to a_m, hence, m will be referring to the cardinality of the input mentions.
- The ensemble of linkable Wikipedia entities of a specific mention text will be denoted as $Pg(a)$.

3.1. Preprocessing, Knowledge Extraction, Transformation, and Load

Our proposed methodologies rely on domain knowledge and rich semantic information extracted from Wikipedia. However, our domain knowledge base is not limited to Wikipedia, but can be extended with semantic information from any Wikipedia annotated text corpora.

Wikipedia is being actively maintained by a large number of administrators and individual contributors, who collaboratively accord encyclopedic articles spanning on a variety of domains as a result of an online public consensus convergence process to textual and conceptual descriptions that can be interpreted as a knowledge ontology. However, apart from an encyclopedic article entity inventory, Wikipedia features a rich internal and external hyperlink structure, interconnecting internal articles or external resources with textual mentions. In this work, we exploit this inventory of mention texts as our set of

candidate entity mentions. More specifically, our mention universe consists of mentions linked to article pages in the *(Main)* MediaWiki namespace with identifier 0 [42]. This main namespace also contains redirect pages, essentially introducing a redirection from a set of article titles to a specific destination entity. We performed a unification to the end Wikipedia Entity ID on our internal representation.

The pre-processing of our Wikipedia snapshot involves common rules and methodologies from similar works [3–5,7–9,34], for managing stop-words and special symbols and characters. We use Unicode format and apply escape rules for special characters along with a common stop-words list included in our codebase. Finally, we extract a 3-tuple inventory for each individual mention occurrence within the Wikipedia main pages corpus, consisting of the:

Mention text: the anchor hyperlink text of the specific mention occurrence.
Mention Wikipedia entity ID: the Wiki ID [43] linked by the *mention text*.
Source article Wikipedia ID: the Wiki ID [43] of the page of occurrence of the specific mention.

As Wikipedia tends to contain some less accurate mentions, with a low frequency, as outlined in previous works [2,8,9], some relative frequency rules for discarding the long tail of the mention's occurrence distribution are common. In our current experimental setup, for further simplifying our methods and densifying the data structures involved, we effectively maintain the top 2 most-common interpretations of each mention text, for the computation of *relative commonness*.

For streamlining the anchor-extraction-process performance and mitigating some of the impediments outlined in [2], a mention inventory of the title mentions to their respective articles has been used for the expansion of our mention set, using a source article Wikipedia ID of 0, for creating our *mentionMap* hash map in memory.

In the relevant literature, *Commonness (1)* [1,6] is a widely used feature and has been a key factor in early approaches and several run-time performance-oriented works.

$$Commonness(p_k, a) = P(p_k|a) \tag{1}$$

The *Commonness* score for each p_k of any Wikipedia anchor occurrence in the Wikipedia corpus can be efficiently pre-calculated by a single parse of the Wikipedia corpus. Specifically, the parse may retain all occurrences of a as an inter-wiki link, along with its linked Wikipedia entities. The count of occurrences of a as a link to a specific Wikipedia entity p_k, divided by the total number of occurrences of a as an inter-wiki link, derives the $Commonness(p_k, a)$ score. This score has been calculated in a map-reduce fashion from our implementation during our initial experimental exploration stages of this work, for the creation of an in-memory hash map for a comparison with the *mentionMap* approach of [2]. Taking this a step further, we introduce *Relative Commonness* of the most common p_k annotation of mention a as:

$$\begin{gathered} Relative\ Commonness(p_j, a) = \frac{|P(p_k|a) - P(p_j|a)|}{P(p_k|a)}, \\ p_k : \max(P(p_k|a),\ p_k \in \{p_1,\ p_2 \dots p_n\})), \\ p_j : \max(P(p_j|a),\ p_j \in \{p_1,\ p_2 \dots p_n\} - \{p_k\})), \end{gathered} \tag{2}$$

For example, let a sequence of candidate detected mentions from a text fragment:

$$\dots\ a_0, a_1, a_2 \dots$$

For:

$|\,Pg(a_0)\,| = 2$, $Pg(a_0) = \{p_{00}, p_{01}\}$ and
$P(p_{00}\,|\,a_0) = 0.8$,
$P(p_{01}\,|\,a_0) = 0.2$,

$$\text{the } Relative\ Commonness(p_{00}, a_0) \text{ would be : } \frac{|0.8 - 0.2|}{0.8} = 0.75.$$

For:
$|\,Pg(a_1)\,| = 3$, $Pg(a_1) = \{p_{10}, p_{11}, p_{12}\}$ and
$P(p_{10}\,|\,a_1) = 0.8$,
$P(p_{11}\,|\,a_1) = 0.1$,
$P(p_{12}\,|\,a_1) = 0.1$,

$$\text{the } Relative\ Commonness(p_{10}, a_1) \text{ would be : } \frac{|0.8 - 0.1|}{0.8} = 0.875.$$

The *relative commonness* measure is effectively expressing the normalized difference of the highest commonness entity, with the second highest commonness entity for a mention. This measure improves the ranking of candidate p_b annotations of an a, as intuitively it is expected that a frequently common candidate p_c is more valuable compared to the next most common candidate p, when the absolute commonness of $Commonness(p_b, a)$ >> $Commonness(p_c, a)$. Our initial commonness-based experimental-exploration analysis was further improved, as intuitively anticipated by the introduction of *relative commonness*; hence, the current contribution is centered around the *relative commonness*-based methodology. To summarize, the *relative commonness* measure is inherently computing and accuracy performant and interpretable.

Relative commonness ranges in [0, 1], is a valuable metric that can be efficiently pre-calculated for the entire set of candidates mentions, and is being employed for our mention disambiguation and confidence evaluation steps. Evidently, in Formula (2), unambiguous mentions have a *relative commonness* of 1. At the last stage of pre-processing, we create a hash map of the mention and the max occurring *relative commonness* value element for the set of mentions encountered as links or titles in Wikipedia.

3.2. Entity Linking

The current work addresses the end-to-end entity linking task with Wikipedia Entities, namely the mention extraction of the semantically dominant concepts from an unstructured text fragment, followed by the disambiguation and annotation to entities drawn by the ontology interpretation of Wikipedia. Our focus is aimed at combining the run-time efficiency with high-quality performance in terms of the predicted entity-linking results, for empowering (i) mobile and edge applications, or (ii) reducing the computational complexity of runtime demanding more accurate approaches, by layering such methods on top of our method's high confidence results, or further processing our fuzzy set derivation methodology. In this section, we present a method attaining state-of-the-art run-time performance compared to [2], while significantly outperforming its quality-performance characteristics. Specifically, Section 3.2.1 describes our mention parsing process, and Section 3.2.2 provides detailed information regarding the subsequent mention disambiguation algorithm. Finally, in Section 3.2.3, we present our confidence evaluation scoring methodologies for the assessment of an outcome mention prediction.

3.2.1. Mention Parsing

The first stage of any end-to-end entity linking approach involves some type of extraction of candidate mentions, bearing semantic relevance within the unstructured text input. This phase is supported by the mention inventory described in Section 3.1. Although this limited inventory outlines the limits of our entity-detection capabilities, the method described in Section 3.1 ensures full coverage of all Wikipedia entities by at least the Wikipedia entity article title, and for most of the entities several distinct mention hyperlink text alternatives drawn from the current evaluation Wikipedia dump [44].

At this stage, the unstructured input text is tokenized, to form n-grams in the size range of one up to six. As pointed out by previous works [1,3,8,9], longer or shorter n-gram size ranges may impose both efficacy and efficiency challenges. The n-grams created, are compared and matched within our *mentionMap*, and a simple rule of preference to longer n-grams is used for the final candidate mention selection for our *relative commonness* method.

The parsing phase is based on highly efficient rules, both in terms of run-time and precision performance, with interpretability in mind.

However, the entity-linking problem tends to convolute to a philosophical question on several occasions, and that is indeed reflected on some inaccurate output annotations we evaluated during our experimentation. For instance, the phrase "England national football team", in some contexts, is linked and interpreted to the "England national football team" titled entity, however, the interpretation of the word "England" to the Wikipedia entity "England" and the word "football" to the homonym entity could be considered as a correct annotation as well. In fact, choosing the right entity link between the two options is challenging even for human interpretation. A focus on more specific entity links would yield the "England national football team" as an entity link, while focusing on broader entities provides an output of entity links in broader and less-specific scope. Both interpretations cannot be considered as entirely inaccurate, hence, maintaining both entity links can effectively increase the value of semantic context output. Consequently, a fuzzy set approach [45,46] may be crucial in reflecting the semantic context in linked Wikipedia Entities. To that end, our *fuzzy relative commonness* method extracts and evaluates the set of mention matches as a fuzzy set.

More specifically, the *fuzzy relative commonness* method, extends the mention parsing phase of the *relative commonness* methodology, using the assumption that a given text input candidate mention representation is not limited to an explicit entity from the Wikipedia entity ensemble. As Wikipedia entity scope is variant, ranging from very broad and generic to very specific and niche concepts, during the fuzzy set mention parsing we maintain all candidates mention matches. For example, in the phrase "*England national football team*", we may extract the following candidate mentions, as a fuzzy set for the interpretation of that phrase: {"*England national football team*", "*England*", "*National*", "*Football*", "*Team*", etc.}. The number of occurrences for each set item can be used for weighing their grade of membership, yet in the context of this approach, the plain assessment of membership is being evaluated during our experiments.

3.2.2. Mention Disambiguation

The run-time performance is at the limelight of this work. Thereby, the computational overhead has been shifted to the creation of data structures capable of supporting fast operations in-memory, for minimizing the run-time complexity. Specifically, as described in Section 3.1, a hash table containing the max *Relative Commonness* values for the mention universe of our method is pre-calculated and loaded in memory, allowing for disambiguation in *O(1)*, on the average case. Both unambiguous and polysemous mentions that have been detected in the mention extraction step are assigned a linked Wikipedia entity. For unambiguous mentions where $|Pg(a)| = 1$, the value of *relative commonness* is 1, while for ambiguous mentions it may range in the (0, 1) interval.

3.2.3. Confidence Evaluation Score

The successful evaluation of confidence for the disambiguation process is crucial for the adoption of our methodology both on the edge and mobile applications domain, and in a phased entity linking approach is paired with a more compute intensive methodology. The confidence evaluation is a core component of the entity linking process and as such performance is yet again a key factor. To that end, the *relative commonness* has been propitious in both the disambiguation and confidence evaluation stages and has been proven to be a great metric in depicting disambiguation quality. As a result, a hash table lookup of *O(1)*, on the average case, is involved in this stage.

3.3. Experimental Evaluation Methodology

For the experimental assessment of our methods, we performed a baseline comparison with the best performing methods described by [2], which according to our literature review attain state-of-the-art run-time performance to date. More specifically, we implemented and

open sourced *RedW* with SR_{norm} scoring method [47], as described in [2], as this approach attains a higher F1 score than most comparative evaluations from the authors.

Our experiments leverage Wikimedia Foundation Wikipedia exports from 20 April 2022 [44]. We processed the snapshot of all Wikipedia article pages from the dump enwiki-20220420-pages-articles.xml.bz2 [48], and the specific resource in WikiText [49] formatting has been, therefore, the input of both our baseline and methodologies implementation for an even comparison based on the same knowledge sources. The ETL process, described in Section 3.1 above, is developed in a distributed processing framework with horizontal scalability in mind, featuring the capacity to follow the future expansion of Wikipedia, while achieving high performance via a streaming-pipeline implementation.

The widely adopted AIDA CoNLL test-a and test-b datasets available in [50] have been used, containing entity annotations from the CoNLL 2003 entity recognition task. The datasets creation is derived by the Reuters Corpus, as described in [51], and has been, therefore, adopted by several related works. Furthermore, we evaluated the popular ClueWeb12-wned [52,53] and WNED-WIKI [53] datasets, for ensuring a thorough experimental assessment of our method. Our evaluation leverages the ETL implementation described in [41] and open-sourced by Facebook Research in [54].

For the quality performance comparative evaluation of our methods, the typical Precision (3), Recall (4), and F1 score (5) metrics have been applied:

$$\text{Precision} = \frac{TP}{TP + FP} \tag{3}$$

$$\text{Recall} = \frac{TP}{|mentions|} \tag{4}$$

$$\text{F1} = 2 \times \frac{Precision * Recall}{Precision + Recall} \tag{5}$$

True Positives (TP) correspond to a successful mention detection and a compatible entity linking output with our evaluation dataset ground truth. False Positives (FP) correspond to either failure to detect the correct mention during the mention parsing phase or failure to derive a correct entity linking output, which is identical with our dataset ground truth entity id.

4. Results and Discussion

The experiments for our evaluation can be practically carried out on almost any modern commodity personal computer. For efficiently pre-processing Wikipedia, we used a 40vCPU instance with 64 GB memory and Apache Spark framework version 3.2.1 [55]. Our implementation is mainly using the PySpark [56] interface for the preprocessing phase and Python 3.8.10 for the actual methods implementation, evaluation, and visualization [57].

In the context of research on the specific domain, several methodology variations have been explored, however, our assessment revealed the highest value on the variants achieving state-of-the-art run-time performance compared to [2], which is a noteworthy precision-performance improvement. Our evaluation explores two methods focusing on different aspects of the problem. The *relative commonness* methodology relies on the calculated data structures from Section 3.1, performing the disambiguation and anchor pruning for the longest-matching token sequence, as described in Section 3.2.1, and the max *attaining relative commonness* mention entity, as detailed in Section 3.2.2. The *fuzzy relative commonness* method uses the same primitives for entity linking all matched token sequences during the anchor parsing phase, as described in Section 3.2.1, in a fuzzy logic approach to the entity linking task.

Our experimental evaluation is covering the end-to-end entity-linking process, including the mention extraction, disambiguation, and confidence-evaluation scoring. According to our analysis, the actual impediment of the end-to-end task is mainly the mentions parsing phase, as also outlined by [2]. The set of Wikipedia entities on the dump utilized after our preprocessing was 11,518,591, and the distinct entities covered by the 17,200,390 distinct

mentions were 6,339,457. As a result, the mention set expansion, described in Section 3.1, enables the full coverage with at least one mention for the set of Wikipedia Entities.

4.1. Experimental Results

The Precision-Recall performance of our methods in comparison with the current state-of-the-art baseline methods is presented in Figure 1 and Table 1. Our fuzzy sets methodology achieves considerably higher performance, especially in the [0, 0.5] recall interval, potentially due to the richer and less-scattered mentions link coverage. However, since the baseline methods approach the problem by selecting a single most-coherent mention, rather than evaluating the full set of available matches, the *fuzzy relative commonness* method can be perceived as depicting the highest attainable end-to-end entity-linking accuracy, based on the available links-domain knowledge in our evaluation Wikipedia dump, for the *relative commonness* entity-linking methodology, considering an ideal mention-extraction phase. Thereby our plain *relative commonness* method performance is our main method for an one-to-one comparison with the highest performing baseline of [2], hence, the fastest accurate state-of-the-art in fast end-to-end Wikification method to date, according to our research. Apparently, on our evaluation datasets our methodology outperforms considerably *RedW* SR_{norm} method across the entire recall range. As seen on Table 1, for full recall, our method attains accuracy near 0.62, compared to the approximately 0.50 accuracy of [2]. For a recall of 0.6, our method attains almost 0.8 precision, compared to approximately 0.57 of [2].

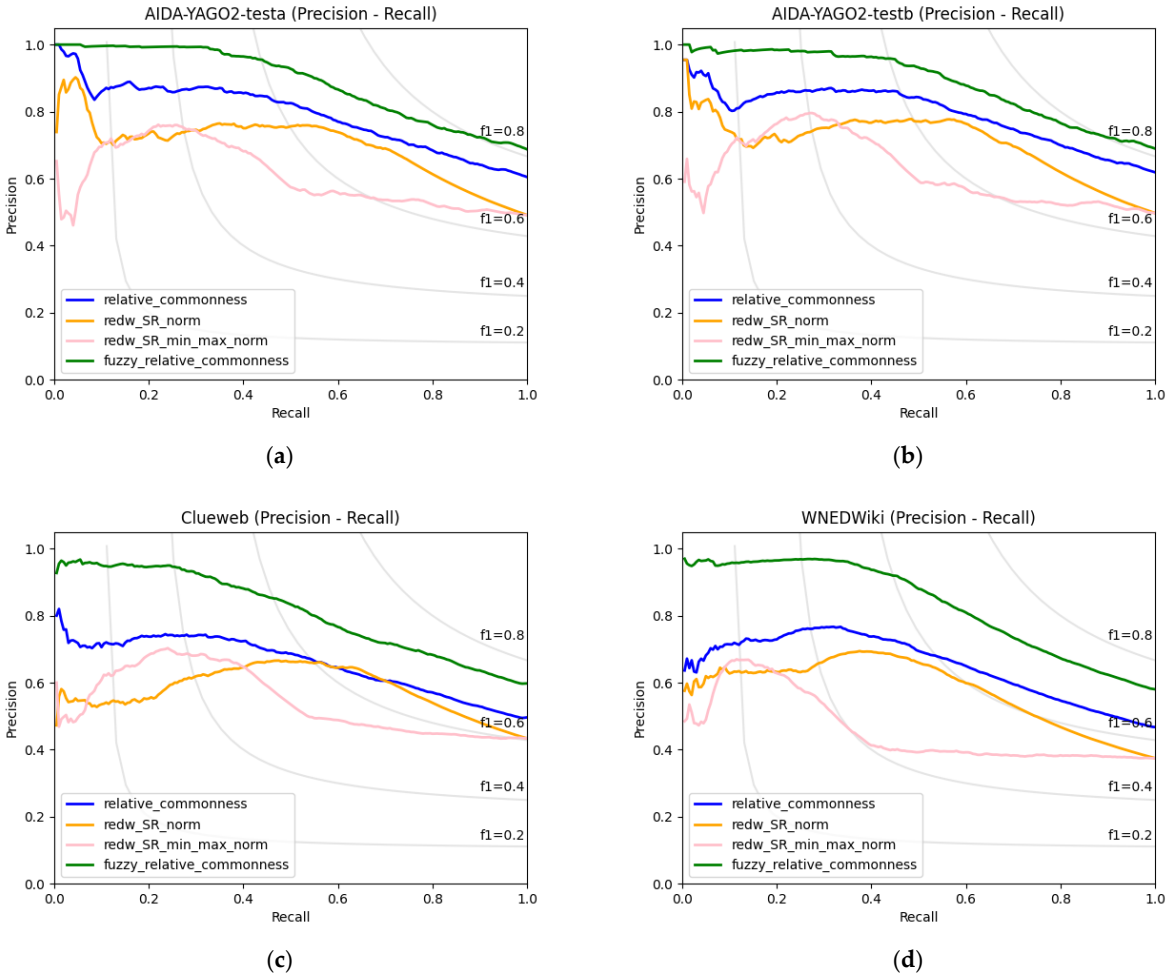

Figure 1. Precision-Recall curves of *Redw* SR_{norm}, *RedW* $SR_{min\ max\ norm}$, *relative commonness*, and *fuzzy relative commonness* for: (**a**) AIDA CoNLL—YAGO test-a dataset [50], (**b**) AIDA CoNLL—YAGO test-b dataset [50], (**c**) ClueWeb12-wned [52,53], and (**d**) WNED-Wiki [53]. F1 lines are included in both diagrams.

Figure 2 and Table 2 display the F1 score performance of our evaluated methodologies, for the evaluated datasets. We should highlight the context variance the evaluation datasets. However, the entity-linking F1 score performance is clearly superior on our introduced methodologies. Specifically, on the [0, 0.4] recall interval, our *relative commonness* method is performing marginally yet consistently better than the baseline methods. The improvement is gradually accelerated in the (0.4, 0.7] interval and far more apparent in the (0.7, 1] recall range. In all cases, the adoption of our simple run-time efficiency-focused methods is proven superior, also from an entity-linking quality perspective, reaching a 10% F1 score improvement in the evaluation datasets in the 1.0 recall case, hence clearly outlining the value of our method for edge and mobile applications.

Table 1. Recall-Precision table of *RedW* SR_{norm}, *RedW* $SR_{min\ max\ norm}$, *relative commonness*, and *fuzzy relative commonness* for AIDA CoNLL—YAGO test-a dataset [50].

Recall	*RedW* SR_{norm}	*RedW* $SR_{min-max}$	*Relative Commonness*	*Fuzzy Relative Commonness*
0.1	0.7411	0.6942	0.8058	0.9799
0.2	0.7246	0.7637	0.8573	0.9844
0.3	0.7517	0.7755	0.8654	0.9792
0.4	0.7670	0.7129	0.8623	0.9638
0.5	0.7707	0.5923	0.8430	0.9318
0.6	0.7640	0.5693	0.7930	0.8777
0.7	0.6964	0.5323	0.7483	0.8235
0.8	0.6193	0.5223	0.6998	0.7659
0.9	0.5508	0.5240	0.6558	0.7227
1.0	0.4972	0.4972	0.6190	0.6899

Recall-Precision table.

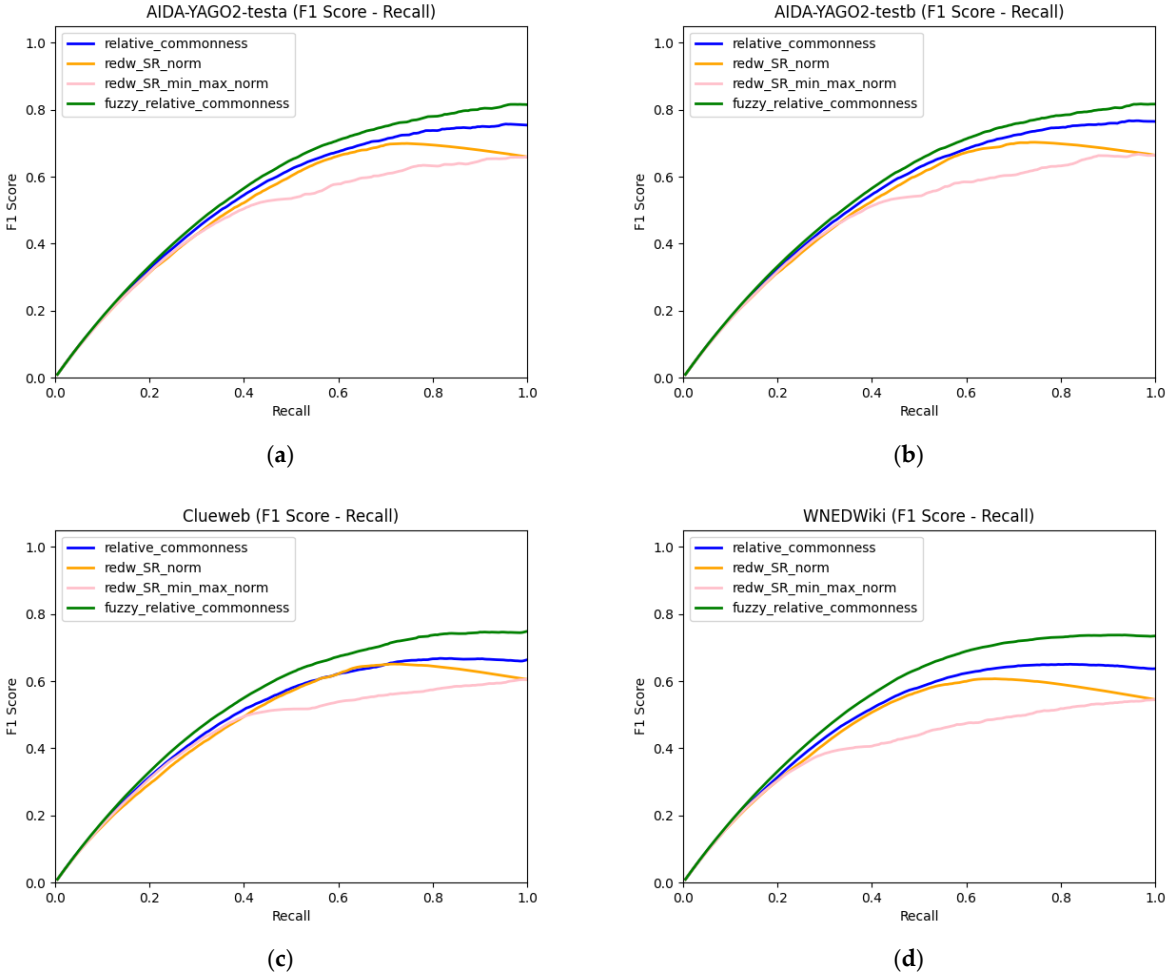

Figure 2. F1 score-Recall curves of *RedW* SR_{norm}, *RedW* $SR_{min\ max\ norm}$, *relative commonness*, and *fuzzy relative commonness* for: (**a**) AIDA CoNLL—YAGO test-a dataset [50], (**b**) AIDA CoNLL—YAGO test-b dataset [50], (**c**) ClueWeb12-wned [52,53], and (**d**) WNED-Wiki [53]. F1 baselines are included in both diagrams.

Table 2. Recall-F1 score table of *RedW* SR_{norm}, *RedW* $SR_{min\ max\ norm}$, *relative commonness*, and *fuzzy relative commonness* for AIDA CoNLL—YAGO test-a dataset [50].

Recall	*RedW* SR_{norm}	*RedW* $SR_{min-max}$	*Relative Commonness*	*Fuzzy Relative Commonness*
0.1	0.1762	0.1748	0.1779	0.1815
0.2	0.3135	0.3170	0.3243	0.3325
0.3	0.4288	0.4326	0.4456	0.4593
0.4	0.5258	0.5125	0.5465	0.5654
0.5	0.6065	0.5423	0.6277	0.6508
0.6	0.6722	0.5842	0.6831	0.7128
0.7	0.6982	0.6048	0.7234	0.7567
0.8	0.6981	0.6320	0.7466	0.7826
0.9	0.6834	0.6624	0.7588	0.8017
1.0	0.6642	0.6642	**0.7646**	0.8165

F1 score-Recall table.

The runtime performance for the end-to-end entity linking process, including the mention-extraction, mention-disambiguation, and confidence-score evaluation of the entire *AIDA CoNLL test-a* and *test-b* b datasets [50], is presented in Table 3. The approach of RedW, according to [2], presents an astounding 95% improvement compared to popular previous baseline approaches such as [8]. Our *relative commonness* method clearly contributes further to those run-time improvements by over 15% for the *AIDA CoNLL test-a* dataset, over 25% for the *CoNLL test-b*, over 13% for the ClueWeb12-wned dataset, and over 12% for the

WNED-WIKI dataset, according to our evaluation, demonstrating a clear improvement in both run-time and entity linking performance, via a simple-yet-effective methodology.

Table 3. Single core run-time table in seconds for AIDA CoNLL test-a and test-b datasets [50].

Dataset	*RedW SR$_{norm}$*	*RedW SR$_{min-max}$*	*Relative Commonness*	*Fuzzy Relative Commonness*
AIDA CoNLL test-a	10.282576	10.245567	**8.561391**	32.421240
AIDA CoNLL test-b	6.675579	6.700371	**4.936130**	18.709551
Clueweb12-wned	85.141593	83.307072	**72.562388**	193.694214
WNED-WIKI	11.431783	11.233509	**9.852915**	37.373979

Timings are in seconds, using Python 3.8.10 [57] *time.time()* function timestamps.

4.2. Future Work

This work leverages context-free features in a simplified yet highly effective methodology, superior to several more compute-intensive and context-aware methodologies, as compared with [2] by the authors. Despite the state-of-the-art performance on both run-time and entity-linking precision in the fast Wikification field, some further improvement areas are worth deeper analysis and experimentation.

Our future analysis can include an experimental comparison of run-time performance against several current popular deep learning alternatives attaining state-of-the-art precision, facilitating the value assessment of the commensurate-benefits question of vast resources allocation to the problem. In addition, a hybrid method leveraging a layered approach for fast entity linking with Wikipedia, for high confidence NED of some mentions using our current methodology, with a resource intensive deep learning approach for the more challenging mentions with higher precision, could improve the run-time performance of the current state-of-the-art general-purpose NED methods.

The Wikification task is heavily reliant on knowledge, hence, the acquisition of semantically linked corpora to Wikipedia entities can further improve our results. To that end, the contribution of unsupervised link-prediction methods on the Wikipedia corpus, along with fuzzy matching for the expansion of inter-wiki coverage, could enrich our knowledge base and further improve our methods accuracy.

Furthermore, the introduction of a parameter for wider or narrower context-mention extraction can further improve the relevant process, in tandem with the introduction of weighting on the Wikipedia title link-enrichment phase.

Finally, the development of a coreference hash table during the pre-processing phase could induce context awareness and encode additional knowledge on our base data structures, for improving our one-shot methodology precision accuracy, with no substantial overhead from a run-time perspective.

5. Conclusions

In this work, we proposed a novel context-free named-entity-disambiguation methodology, achieving state-of-the-art run-time performance by approximately 15%, compared to previous approaches in the fast end-to-end entity linking with a Wikipedia task [2], and up to 10% in entity-linking-accuracy performance in the context of these levels of run-time performance. We introduced the *relative commonness* measure, in a methodology leveraging this feature in a robust data structure for the mention extraction, entity linking, and confidence evaluation of entity-linking outputs for unstructured texts to Wikipedia entities. In addition, we leveraged the *relative commonness* measure for proposing a fuzzy sets representation for the Named Entity disambiguation, for overcoming some of the shortcomings of the mention selection process.

Our highly effective and efficient methodologies can be leveraged due to their reduced computational requirement footprint in edge, mobile, and real-time applications, but can be also leveraged for achieving a vast complexity reduction in conjunction with more complex, precision-oriented, and compute-intensive approaches in a layered architecture, for retrieving, at the first stage, entity links using our *relative-commonness*-based

methodology, followed by a more complex yet accurate entity-linking phase using more compute-intensive methodologies. However, the value of our simple-yet-effective methodology is clearly outlined by our experimental results.

The codebase of our work has been open-sourced in [58], for streamlining the adoption of high-performance semantic representation from AI applications and facilitating further improvements. We intend to explore areas for further enhancing the entity-linking performance of our methodology and expose a public REST API interface for interaction with the *relative commonness* method framework.

The focus of this article has been on proposing and evaluating a new method for reducing the current computational barrier for employing a named entity disambiguation task. Our experiments on established datasets outline propitious results, constituting our method being auspicious for wide adoption on big data.

Author Contributions: Conceptualization, M.A.S. and C.M.; data curation, M.A.S.; formal analysis, M.A.S.; funding acquisition, C.M.; investigation, M.A.S.; methodology, M.A.S.; project administration, M.A.S. and C.M.; resources, M.A.S.; software, M.A.S.; supervision, C.M.; validation, M.A.S.; visualization, M.A.S.; writing—original draft, M.A.S.; writing—review and editing, M.A.S. and C.M. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Data Availability Statement: EnWiki dump of 20,220,420 pages–articles is available online via: https://dumps.wikimedia.org/enwiki/20220420/enwiki-20220420-pages-articles.xml.bz2 (accessed on 20 July 2022). AIDA CoNLL—YAGO dataset is available online via: http://resources.mpi-inf.mpg.de/yago-naga/aida/download/aida-yago2-dataset.zip (accessed on 20 July 2022). The source code of the baseline methodology implementation is available via [47]. The source code of our method implementation, evaluation, and visualizations is available via [58].

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Mihalcea, R.; Csomai, A. Wikify! In Proceedings of the Sixteenth ACM Conference on Conference on Information and Knowledge Management—CIKM '07, Lisbon, Portugal, 6–10 November 2007; ACM Press: New York, NY, USA, 2007; p. 233. [CrossRef]
2. Shnayderman, I.; Ein-Dor, L.; Mass, Y.; Halfon, A.; Sznajder, B.; Spector, A.; Katz, Y.; Sheinwald, D.; Aharonov, R.; Slonim, N. Fast End-to-End Wikification. *arXiv* **2019**. [CrossRef]
3. Milne, D.; Witten, I.H. Learning to link with wikipedia. In Proceedings of the 17th ACM Conference on Information and Knowledge Mining—CIKM '08, Napa Valley, CA, USA, 26–30 October 2008; ACM Press: New York, NY, USA, 2008; p. 509. [CrossRef]
4. Makris, C.; Simos, M.A. Novel Techniques for Text Annotation with Wikipedia Entities. In Proceedings of the 10th IFIP WG 12.5 International Conference, AIAI 2014, Rhodes, Greece, 19–21 September 2014; pp. 508–518. [CrossRef]
5. Kulkarni, S.; Singh, A.; Ramakrishnan, G.; Chakrabarti, S. Collective annotation of Wikipedia entities in web text. In Proceedings of the 15th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining—KDD '09, Paris, France, 28 June–1 July 2009; ACM Press: New York, NY, USA, 2009; p. 457. [CrossRef]
6. Cucerzan, S. Large-Scale Named Entity Disambiguation Based on Wikipedia Data. In Proceedings of the 2007 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning (EMNLP-CoNLL), Prague, Czech Republic, 28–30 June 2007; Association for Computational Linguistics: Prague, Czech Republic, 2007; pp. 708–716.
7. Milne, D.; Witten, I.H. An Effective, Low-Cost Measure of Semantic Relatedness Obtained from Wikipedia Links. In Proceedings of the AAAI 2008, Chicago, IL, USA, 13–17 July 2008.
8. Ferragina, P.; Scaiella, U. TAGME. In Proceedings of the 19th ACM International Conference on Information and Knowledge Management—CIKM '10, Toronto, ON, Canada, 26–30 October 2010; ACM Press: New York, NY, USA, 2010; p. 1625. [CrossRef]
9. Piccinno, F.; Ferragina, P. From TagME to WAT. In Proceedings of the First International Workshop on Entity Recognition & Disambiguation—ERD '14, Gold Coast, Queensland, Australia, 11 July 2014; ACM Press: New York, NY, USA, 2014; pp. 55–62. [CrossRef]
10. Chai, J.; Zeng, H.; Li, A.; Ngai, E.W.T. Deep learning in computer vision: A critical review of emerging techniques and application scenarios. *Mach. Learn. Appl.* **2021**, *6*, 100134. [CrossRef]
11. Chen, L.; Li, S.; Bai, Q.; Yang, J.; Jiang, S.; Miao, Y. Review of Image Classification Algorithms Based on Convolutional Neural Networks. *Remote Sens.* **2021**, *13*, 4712. [CrossRef]
12. Yoon, S.-H.; Yu, H.-J. A Simple Distortion-Free Method to Handle Variable Length Sequences for Recurrent Neural Networks in Text Dependent Speaker Verification. *Appl. Sci.* **2020**, *10*, 4092. [CrossRef]

13. Trinh Van, L.; Dao Thi Le, T.; le Xuan, T.; Castelli, E. Emotional Speech Recognition Using Deep Neural Networks. *Sensors* **2022**, *22*, 1414. [CrossRef] [PubMed]
14. Lee, M.; Chang, J.-H. Augmented Latent Features of Deep Neural Network-Based Automatic Speech Recognition for Motor-Driven Robots. *Appl. Sci.* **2020**, *10*, 4602. [CrossRef]
15. Raghavan, P.; Gayar, N. el Fraud Detection using Machine Learning and Deep Learning. In Proceedings of the 2019 International Conference on Computational Intelligence and Knowledge Economy (ICCIKE), Dubai, United Arab Emirates, 11–12 December 2019; pp. 334–339. [CrossRef]
16. Jang, H.-J.; Cho, K.-O. Applications of deep learning for the analysis of medical data. *Arch. Pharmacal. Res.* **2019**, *42*, 492–504. [CrossRef] [PubMed]
17. Suzuki, K. Overview of deep learning in medical imaging. *Radiol. Phys. Technol.* **2017**, *10*, 257–273. [CrossRef] [PubMed]
18. Pandey, B.; Kumar Pandey, D.; Pratap Mishra, B.; Rhmann, W. A comprehensive survey of deep learning in the field of medical imaging and medical natural language processing: Challenges and research directions. *J. King Saud Univ. Comput. Inf. Sci.* **2021**, *in press*. [CrossRef]
19. Grigorescu, S.; Trasnea, B.; Cocias, T.; Macesanu, G. A survey of deep learning techniques for autonomous driving. *J. Field Robot.* **2020**, *37*, 362–386. [CrossRef]
20. Young, T.; Hazarika, D.; Poria, S.; Cambria, E. Recent Trends in Deep Learning Based Natural Language Processing [Review Article]. *IEEE Comput. Intell. Mag.* **2018**, *13*, 55–75. [CrossRef]
21. Makris, C.; Simos, M.A. OTNEL: A Distributed Online Deep Learning Semantic Annotation Methodology. *Big Data Cogn. Comput.* **2020**, *4*, 31. [CrossRef]
22. Murdoch, W.J.; Singh, C.; Kumbier, K.; Abbasi-Asl, R.; Yu, B. Definitions, methods, and applications in interpretable machine learning. *Proc. Natl. Acad. Sci. USA* **2019**, *116*, 22071–22080. [CrossRef] [PubMed]
23. Chakraborty, S.; Tomsett, R.; Raghavendra, R.; Harborne, D.; Alzantot, M.; Cerutti, F.; Srivastava, M.; Preece, A.; Julier, S.; Rao, R.M.; et al. Interpretability of deep learning models: A survey of results. In Proceedings of the 2017 IEEE SmartWorld, Ubiquitous Intelligence & Computing, Advanced & Trusted Computed, Scalable Computing & Communications, Cloud & Big Data Computing, Internet of People and Smart City Innovation (SmartWorld/SCALCOM/UIC/ATC/CBDCom/IOP/SCI), San Francisco, CA, USA, 4–8 August 2017; pp. 1–6. [CrossRef]
24. Thompson, N.C.; Greenewald, K.; Lee, K.; Manso, G.F. The Computational Limits of Deep Learning. *arXiv* **2020**. [CrossRef]
25. Navigli, R. Word sense disambiguation. *ACM Comput. Surv.* **2009**, *41*, 1–69. [CrossRef]
26. Scarlini, B.; Pasini, T.; Navigli, R. Sense-Annotated Corpora for Word Sense Disambiguation in Multiple Languages and Domains. In Proceedings of the 12th Language Resources and Evaluation Conference, Marseille, France, 11–16 May 2020; European Language Resources Association: Marseille, France, 2020; pp. 5905–5911.
27. Pasini, T. The Knowledge Acquisition Bottleneck Problem in Multilingual Word Sense Disambiguation. In Proceedings of the Twenty-Ninth International Joint Conference on Artificial Intelligence, Yokohama, Japan, 7–15 January 2021; pp. 4936–4942. [CrossRef]
28. Goodfellow, I.J.; Mirza, M.; Xiao, D.; Courville, A.; Bengio, Y. An Empirical Investigation of Catastrophic Forgetting in Gradient-Based Neural Networks. *arXiv* **2013**. [CrossRef]
29. Sil, A.; Kundu, G.; Florian, R.; Hamza, W. Neural cross-lingual entity linking. In Proceedings of the Thirty-Second AAAI Conference on Artificial Intelligence and Thirtieth Innovative Applications of Artificial Intelligence Conference and Eighth AAAI Symposium on Educational Advances in Artificial Intelligence, New Orleans, LA, USA, 2–7 February 2018; pp. 5464–5472.
30. Gmach, D.; Chen, Y.; Shah, A.; Rolia, J.; Bash, C.; Christian, T.; Sharma, R. Profiling Sustainability of Data Centers. Proceedings of 2010 IEEE International Symposium on Sustainable Systems and Technology, Arlington, VA, USA, 17–19 May 2010; pp. 1–6. [CrossRef]
31. Freitag, C.; Berners-Lee, M.; Widdicks, K.; Knowles, B.; Blair, G.; Friday, A. The climate impact of ICT: A review of estimates, trends and regulations. *arXiv* **2021**. [CrossRef]
32. Gale, W.A.; Church, K.W.; Yarowsky, D. A method for disambiguating word senses in a large corpus. *Comput. Humanit.* **1992**, *26*, 415–439. [CrossRef]
33. Hoffart, J.; Yosef, M.A.; Bordino, I.; Fürstenau, H.; Pinkal, M.; Spaniol, M.; Taneva, B.; Thater, S.; Weikum, G. Robust Disambiguation of Named Entities in Text. In Proceedings of the Conference on Empirical Methods in Natural Language Processing, Edinburgh, UK, 27–31 July 2011; pp. 782–792.
34. Han, X.; Sun, L.; Zhao, J. Collective entity linking in web text. In Proceedings of the 34th International ACM SIGIR Conference on Research and Development in Information—SIGIR '11, Beijing, China, 24–28 July 2011; ACM Press: New York, NY, USA, 2011; p. 765. [CrossRef]
35. Sun, Y.; Lin, L.; Tang, D.; Yang, N.; Ji, Z.; Wang, X. Modeling Mention, Context and Entity with Neural Networks for Entity Disambiguation. In Proceedings of the 24th International Joint Conference on Artificial Intelligence, Buenos Aires, Argentina, 25–31 July 2015; pp. 1333–1339.
36. Yamada, I.; Shindo, H.; Takeda, H.; Takefuji, Y. Joint Learning of the Embedding of Words and Entities for Named Entity Disambiguation. In Proceedings of the 20th SIGNLL Conference on Computational Natural Language Learning, Berlin, Germany, 11–12 August 2016; Association for Computational Linguistics: Stroudsburg, PA, USA, 2016; pp. 250–259. [CrossRef]

37. Ganea, O.-E.; Hofmann, T. Deep Joint Entity Disambiguation with Local Neural Attention. In Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing, Copenhagen, Denmark, 9–11 September 2017; Association for Computational Linguistics: Stroudsburg, PA, USA, 2017; pp. 2619–2629. [CrossRef]
38. Le, P.; Titov, I. Improving Entity Linking by Modeling Latent Relations between Mentions. In Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), Dublin, Ireland, 22–27 May 2022; Association for Computational Linguistics: Stroudsburg, PA, USA, 2018; pp. 1595–1604. [CrossRef]
39. Radhakrishnan, P.; Talukdar, P.; Varma, V. ELDEN: Improved Entity Linking Using Densified Knowledge Graphs. In Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers), New Orleans, LA, USA, 1–6 June 2018; Association for Computational Linguistics: Stroudsburg, PA, USA, 2018; pp. 1844–1853. [CrossRef]
40. Fang, Z.; Cao, Y.; Li, Q.; Zhang, D.; Zhang, Z.; Liu, Y. Joint Entity Linking with Deep Reinforcement Learning. In Proceedings of the The World Wide Web Conference on—WWW '19, San Francisco, CA, USA, 13–17 May 2019; ACM Press: New York, NY, USA, 2019; pp. 438–447. [CrossRef]
41. Wu, L.; Petroni, F.; Josifoski, M.; Riedel, S.; Zettlemoyer, L. Scalable Zero-shot Entity Linking with Dense Entity Retrieval. *arXiv* **2019**. [CrossRef]
42. MediaWiki/Help:Namespaces. Available online: https://MediaWiki/Help:Namespaces (accessed on 25 April 2022).
43. MediaWiki:Wiki_ID. Available online: https://www.mediawiki.org/wiki/Manual:Wiki_ID (accessed on 25 April 2022).
44. EnWiki Dump 20220420. Available online: https://dumps.wikimedia.org/mkwiki/20220420/ (accessed on 25 April 2022).
45. Zadeh, L.A. From computing with numbers to computing with words. From manipulation of measurements to manipulation of perceptions. *IEEE Trans. Circuits Syst. I Fundam. Theory Appl.* **1999**, *46*, 105–119. [CrossRef]
46. Zadeh, L.A. Fuzzy sets. *Inf. Control* **1965**, *8*, 338–353. [CrossRef]
47. RedW CodeBase. Available online: https://github.com/mikesimos/redw (accessed on 25 May 2022).
48. EnWiki Dump 20220420 Pages-Articles. Available online: https://dumps.wikimedia.org/mkwiki/20220420/mkwiki-20220420-pages-articles.xml.bz2 (accessed on 25 April 2022).
49. Specs/wikitext/1.0.0 MediaWiki. Available online: https://www.mediawiki.org/wiki/Specs/wikitext/1.0.0 (accessed on 25 April 2022).
50. AIDA CoNLL-YAGO Dataset. Available online: http://resources.mpi-inf.mpg.de/yago-naga/aida/download/aida-yago2-dataset.zip (accessed on 25 April 2022).
51. Tjong Kim Sang, E.F.; de Meulder, F. Introduction to the CoNLL-2003 shared task. In Proceedings of the Seventh Conference on Natural Language Learning at HLT-NAACL 2003, Edmonton, AB, Canada, 31 May–1 June 2003; Association for Computational Linguistics: Morristown, NJ, USA, 2003; pp. 142–147. [CrossRef]
52. The ClueWeb12 Dataset. Available online: https://lemurproject.org/clueweb12/ (accessed on 20 July 2022).
53. Guo, Z.; Barbosa, D. Robust named entity disambiguation with random walks. *Semant. Web* **2018**, *9*, 459–479. [CrossRef]
54. BLINK Source Code. Available online: https://github.com/facebookresearch/BLINK (accessed on 25 April 2022).
55. Spark. Available online: https://spark.apache.org/downloads.html (accessed on 25 April 2022).
56. PySpark. Available online: https://spark.apache.org/docs/3.2.1/api/python/ (accessed on 25 April 2022).
57. Python 3.8.10. Available online: https://www.python.org/downloads/release/python-3810/ (accessed on 25 April 2022).
58. Methods CodeBase. Available online: https://github.com/mikesimos/fast-wikification (accessed on 25 May 2022).