AGNUS: Robust Entity Disambiguation using LLMs

Anonymous ACL submission

Abstract

Entity disambiguation (ED) is the process of disambiguating entities relating to a knowledge base and a necessary step of the entity linking workflow. With the advent of pretrained generative large language models (LLM), the field of natural language processing has been revolutionised, yet related techniques for ED are scarce. In this paper, we introduce AGNUS an approach leveraging pretrained LLM contextual knowledge to disambiguate entities. We mitigate challenges posed by modern LLMs: order-dependant bias for candidate options, hallucinations and evaluation data contamination. We reach state-of-the-art results in 4 datasets, beating prior work by 3.7% on average for zero-shot configurations, provide code and a novel synthetic dataset for entity disambiguation[1].

1 Introduction

Entity disambiguation – choosing an entity among candidates for a textual mention and given context – remains a critical challenge for semantic web applications and text analysis to this day. While Large Language Models (LLMs) have transformed the field of Natural Language Processing (NLP), as of writing their application to entity disambiguation has remained limited. Applying LLMs naively to Entity Disambiguation (ED) entails following issues:

- **Order Bias**: LLMs exhibit token order-dependent predictions, with performance varying up to 14% in our experiments across candidate permutations, decreasing disambiguation robustness.
- **Hallucinations**: LLMs may suggest options not within a designated candidate set.

As such, we consider naive approaches unfit for robust disambiguation, leading to our research question:

How can LLMs disambiguate entities in a robust fashion?

Particularly as research involving reasoning capabilities of LLMs for task completion is starting to bear fruit, the context-dependant task of entity disambiguation could greatly benefit from their use.

We present AGNUS , a novel framework addressing these challenges through:

- **Masked Attention Candidate Set**: Order-invariant encoding through position embedding overloading.
- **Agnus Contextual Decoding**: We constrain generated tokens to valid candidates in an auto-regressive tree-based fashion.

Our approach to encoding each entity candidate into an order-invariant collection prevents interactions from one candidate to any other. By preventing transformer-based LLMs from applying attention between entities within a given candidate set and overloading their position embeddings, we effectively hide the candidate order from LLMs altogether.

We restrict LLM text generation in a tree-based autoregressive fashion, eliminating disambiguation hallucinations while allowing for LLM reasoning capabilities via decoding strategies choose contextually optimal entities from a given candidate space.

In this paper, we apply LLMs with loosely-structured entity-specific representation criteria, disregarding entity connectivity aspects within Knowledge Graphs (KGs). Further, as disambiguation criteria for our method may be loosely-defined, one can attribute different types of information to entities (e.g. entity types, labels, description)

[1] https://anonymous.4open.science/r/Agnus/README.md

from a variety of sources, allowing for easy out-of-the-box disambiguation from custom, possibly incoherently-connected or even mixed Knowledge Bases (KBs) without the need for training or structural changes. Finally, as LLMs make use of large quantities of oftentimes unspecified data for their training, it may cause issue for meaningful evaluation due to potential benchmark contamination. In an attempt to mitigate contamination, we introduce a method for generating synthetic evaluation data.

Our experiments demonstrate AGNUS achieves:

- Elimination of order bias, removing position-induced variations (Section 3.2).
- Prohibits hallucinations via autoregressive output restrictions (Section 3.3).
- Reduces benchmark contamination by creating a novel synthetic dataset (Section 4.4) and introducing a synthetic dataset generation approach for entity linking and entity disambiguation.

Our contributions advance the state-of-the-art for ED by introducing:

- Robust zero-shot ED pipeline consisting of:
  - Masked Attention Candidate Set (MACS) encoding for candidate order-invariant disambiguation.
  - Agnus Contextual Decoding (ACDC) to constrain LLM responses to valid ones.
- Contamination-resistant evaluation methodology with flexible contextual criteria.
- Open-source implementation[2] for out-of-the-box zero-shot disambiguation.

In the following, we introduce related work for entity disambiguation (Section 2.1) and large language models (Section 2.2).

2 Related Work

2.1 Entity Disambiguation

Entity disambiguation (ED) is a critical task in natural language processing and understanding, where the goal is to map ambiguous entity mentions in text to their correct entries in a knowledge base. Current state-of-the-art ED and entity linking models (Shavarani and Sarkar, 2023; van Hulst et al., 2020; Barba et al., 2022; Ding et al., 2024a; Xiao et al., 2023; Ayoola et al., 2022; Orlando et al., 2024) make use of various deep learning architectures to outperform more traditional works. In recent years, transformer-based systems, such as BLINK (Wu et al., 2020), REL (van Hulst et al., 2020), SpEL (Shavarani and Sarkar, 2023), DeepType (Raiman and Raiman, 2018) and GENRE (Cao et al., 2021) have taken over the stage with many basing themselves on BERT (Devlin et al., 2019) embeddings. Recently, LLM-based systems have entered the space with ChatEL (Ding et al., 2024b) and EntGPT (Ding et al., 2024a). In (Ding et al., 2024a), authors improve entity disambiguation over naive LLM baselines by tackling the issues with prompt engineering and providing LLM backbones with self-generated contextual data.

2.2 Large Language Models

Applying LLMs to ED is accompanied by a multitude of considerations when contrasted with more traditional ED. Among these, there exist benchmark contamination (Section 2.2.1), hallucinations (Section 2.2.2), decoding mechanisms (Section 2.2.3) and order-specific biases (Section 2.2.4) that endanger robust disambiguation. In the following, we address these areas of prior work.

2.2.1 Dataset Contamination

Benchmark contamination in LLMs (Xu et al., 2024) has become a critical issue as models trained on vast amounts of publicly available data may inadvertently 'memorize' aspects of popular benchmark datasets, potentially leading to inflated estimates of their true capabilities.

To address these challenges, researchers have started developing various countermeasures (Chen et al., 2025), including dynamic evaluation benchmarks (Wang et al., 2025; Zhu et al., 2024a,b) to effectively prevent pre-benchmarking disclosure. Another measure is to provide a means of evaluation for the degree of contamination (Xu et al., 2024) by computing perplexity (Li, 2023) – by applying the exponential function to the average negative log likelihood over a particular sequence of text to measure a model's '*surprise*' (or inverse confidence) for a particular output.

2.2.2 Hallucinations

Despite remarkable capabilities in generating human-like text, LLMs may produce factual inaccu-

[2] https://anonymous.4open.science/r/Agnus/README.md

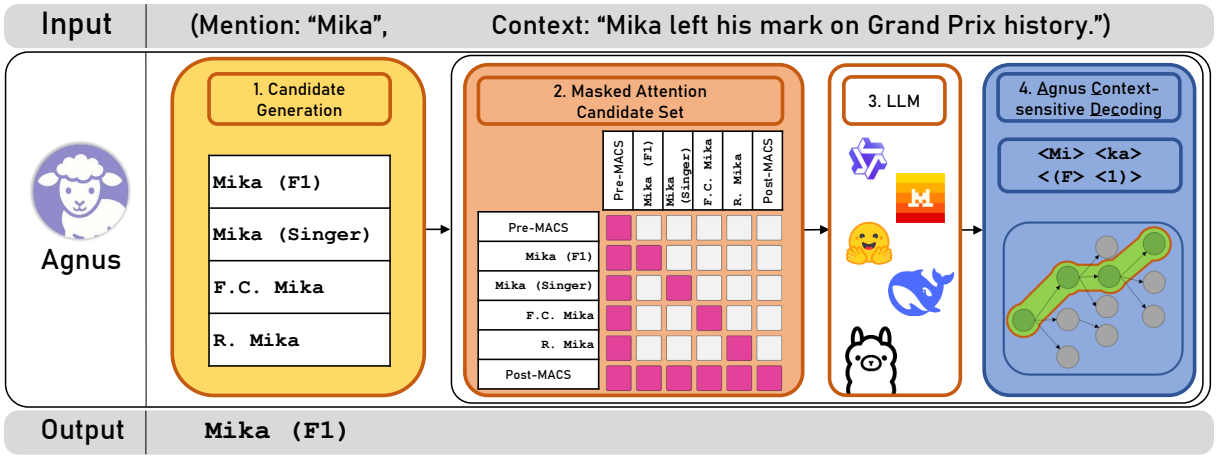

Figure 1: AGNUS disambiguation – Takes an input document, (1.) generates candidate entities for mentions (e.g. MIKA), (2.) applies masked attention to candidate entity collection (MACS, Section 3.2) and (3.) passes representation to a specified LLM, followed by (4.) constrained decoding (ACDC, Section 3.3) for context-sensitive disambiguation and returns the disambiguated entity (e.g. `Mika (F1)`).

racies or nonsensical sequences, a phenomenon referred to as *hallucination* (Huang et al., 2025). The underlying causes of hallucinations are an active area of research. Some potential contributing factors include the vast scale of the training data, potentially containing potentially noisy data (Petroni et al., 2021; Ji et al., 2023) and the autoregressive nature of text generation based on prior tokens (Holtzman et al., 2020; Maynez et al., 2020). The presence of hallucinations poses a significant challenge for the reliable application of LLMs on downstream NLP tasks, posing issue for robust and trustworthy ED. Recent research efforts have started counteracting hallucinations through retrieval augmentation, fact verification and the incorporation of knowledge graphs (Lewis et al., 2020; Pusch and Conrad, 2024).

In this paper, we eliminate the possibility for entity candidate hallucinations by defining a specialised constrained decoding strategy for ED.

2.2.3 Constrained Decoding

Early work on LLMs (Brown et al., 2020; Radford et al., 2019) demonstrated that decoder-only language models process natural language prompts effectively without an enforced schema, meaning that input-output pairs are structurally not bound by predefined templates or grammars. This flexibility allows for broad applicability but introduces challenges in reliability, consistency, and controllability (Bender et al., 2021).

To mitigate challenges of unstructured interaction, researchers have developed various prompt engineering methods (Sahoo et al., 2024; Ouyang et al., 2022a; Madaan et al., 2023; Wei et al., 2022) to implicitly guide LLMs towards more structured outputs. However, these approaches depend on the model's ability to infer structure from textual cues rather than enforcing it. Therefore, constrained decoding (Beurer-Kellner et al., 2024) approaches to enforce strict restrictions on LLM text generation have been developed.

2.2.4 Order Bias

Prior work has established that modern generative large language models demonstrate inherent tendencies toward positional preferences when processing ordered lists of candidate answers (Pezeshkpour and Hruschka, 2023; Wei et al., 2024; Zheng et al., 2023; Anonymous, 2025) and being sensitive towards the arrangement order of otherwise identical answer collections (Dominguez-Olmedo et al., 2023; Li et al., 2023; Li and Gao, 2024; Wang et al., 2023, 2024a; Xue et al., 2024). Approaches to mitigation include compensation for positional preferences (Wei et al., 2024; Zhao et al., 2021), systematic permutation averaging and applying multiple forward passes with varied option sequences (Pezeshkpour and Hruschka, 2023; Wang et al., 2023), as well as reasoning-enhanced strategies (Wang et al., 2024a,b) to attenuate sequence dependence. AGNUS employs a method to mitigate candidate order bias without requiring additional training by adapting the approach from (Anonymous, 2025) to entity disambiguation.

3 AGNUS

In this section we introduce AGNUS , our proposed approach for LLM-based robust entity disambiguation. AGNUS removes order-based bias from entity candidate disambiguation by introducing *Masked Attention Candidate Set* (Section 3.2) based on (Anonymous, 2025) and inhibits LLM hallucinations by applying *Agnus Contextual Decoding* (Section 3.3). In Figure 1, we present AGNUS: from generating entity candidates, applying masked attention (MACS), constrained decoding (ACDC) to final disambiguation for the input document "*Mika left his mark on Grand Prix history.*" and entity mention *Mika*, yielding contextually disambiguated entity `Mika (F1)`[3].

3.1 Disambiguation Setup

AGNUS represents an approach leveraging LLMs for the task of disambiguating entities based on entity candidate information while mitigating LLM-specific challenges. For disambiguation, AGNUS takes as input a document providing context, a mention and a collection of candidate entities generated via pre-existing candidate generation approaches.

Due to leveraging the contextual disambiguation capabilities of LLMs, AGNUS does not require candidate entities to solely be a knowledge base-backed IRI. Instead, candidate entity representation may additionally take any identifying or meaningful form, such as a description, label, type or combination thereof. For each mention contained within an input document, we generate a fixed candidate set (Fig. 1, *Step 1*), employing candidates generated with DBpedia Lookup[4]. Each candidate collection is encoded using MACS (Fig. 1, *Step 2*), embedded into its original textual encoding with parts surrounding it (`Pre-MACS` and `Post-MACS`) being encoded in LLM-specific fashion (see Fig. 3). Subsequently, the resulting encoded prompt is transmitted as a whole to the LLM for contextual parsing (Fig. 1, *Step 3*) and decoded via ACDC (Fig. 1, *Step 4*).

3.2 Masked Attention Candidate Set

Text sequences encoded on modern generative language models rely on underlying position-influenced attention mechanisms and positional embeddings to add a signal for the order of token appearance within a sequence (Anonymous, 2025). This affects desiredly order-invariant sequences, such as candidate collections – an undesirable property for entity disambiguation. To render an LLM order-agnostic for parts of a sequence, we tackle both aspects: mask the attention mechanism between entity candidates (Section 3.2.1) and modify positional embedding values (Section 3.2.2) for candidate entities to simulate similar positions.

In Figure 4 and Table 1, by disambiguating candidate entities across iterations of random candidate shuffles, results without MACS vary depending on candidate entity order and applying MACS eliminates the order-based influence.

3.2.1 Causal Mask

To encode a collection of entity candidates in an order-invariant fashion to the underlying LLM, we apply an adapted version of the commonly-employed triangular attention matrix as causal mask (see Figure 3). Entities within a collection cannot attend to one another (grey entries), but do attend (pink entries) – and are attended to – in otherwise usual LLM fashion to their own prior tokens (diagonal entries) and rest of the token sequence (to PRE-MACS and by POST-MACS). This means that tokens within each candidate's representation continue attending to each other.

3.2.2 Positional Embedding

Every sequence of tokens is attributed a certain range of positional embedding values within its LLM-encoded representation. Within a MACS-encoded collection, every token making up an entity candidate is modified to appear as sharing a similar range of positions (see visualization Fig. 2) as other candidates to the underlying LLM.

To do so, we formally define relative position $i \in [0,..,n_{c_j}-1]$ of each token $t_{c_j,i}$ for entity candidate representation $c_j \in C$ s.t. n_{c_j} is the number of tokens for entity c_j and collection of all candidate entities C for a given mention and T_{c_j} the set of all tokens for c_j: $\forall\, t_{c_j,i} : i \in [0,..,\mathbf{max}_{c \in C}(|n_c - 1|)]$.

Therefore as visualised in Figure 2, the shared range of possible positional embeddings is defined by the token-wise longest candidate within a MACS collection and starts for each candidate at the end of prior sequence's token (PRE-MACS) and afterwards continues the candidate encoding with the succeeding sequence's (POST-MACS) first positional embedding.

[3] https://en.wikipedia.org/wiki/Mika_HÃd'kkinen

[4] https://github.com/dbpedia/dbpedia-lookup

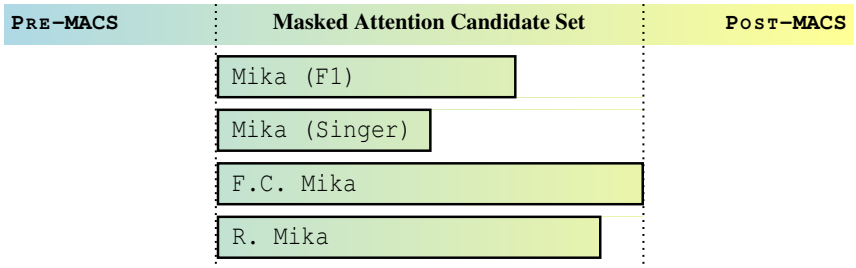

Figure 2: MACS – Positional embedding adaptation: Each candidate entity entry is encoded as being on the same positions for the length of their contents. Candidate entity entries' first positional embedding is treated and encoded analogously for each entry. Post-MACS starting positional embedding is computed as subsequent to the longest option contained within MACS entries.

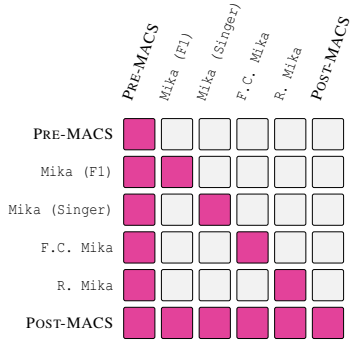

Figure 3: MACS – Causal mask: Example from Fig. 1 for entity candidate representations for entities "Mika (F1)", "Mika (Singer)", "F.C. Mika", "R. Mika". Grey cells signify blocked attention whereas pink signifies enabled attention. Intra entity attention and attention from tokens preceding (Pre-MACS) and succeeding (Post-MACS) MACS is preserved normally s.t. subsequent tokens attend to prior ones.

3.3 Agnus Contextual Decoding

LLMs may add or remove information in unexpected fashions. This ranges from a corrupt expected result format to hallucinating non-existing options. Due to the nature of entity disambiguation, only given options may be produced. As such, we define an input-flexible grammar based on entity candidates. Let the set of candidate sequences be $O = \{o_1, \ldots, o_n\}$ where each candidate option $o_i \in \Sigma^{l_i}$ is a sequence of length l_i. The vocabulary is defined as $\Sigma = \{t_k^i \mid i \in \{1, \ldots, n\},\ k \in \{1, \ldots, l_i\}\} \cup \{\texttt{EOS}\}$. We then define the set of nonterminals as $V = \{X_i^k \mid i \in \{1, \ldots, n\},\ k \in \{0, \ldots, l_i\}\}$ where X_i^k denotes the state after generating the first k tokens of candidate c_i. The start symbol transitions to the initial state of each candidate: $S \to X_1^0 \mid X_2^0 \mid \ldots \mid X_n^0$. For each o_i, we define the following transitions: $X_i^k \to t_{k+1}^i X_i^{k+1}, \quad \forall\, k \in \{0, \ldots, l_i - 1\}, \quad X_i^{l_i} \to \texttt{EOS}$.

4 Experiments and Results

AGNUS combines techniques to create an LLM-enabled approach to robust entity candidate disambiguation. In this section, we conduct experiments to evaluate AGNUS with different configurations regarding representations of entity candidates, LLMs, our candidate encoding (MACS) and our constrained decoding (ACDC). We report entity disambiguation results in comparison to prior work in Table 1.

4.1 Technical Details

All our experiments were run on a server with NVIDIA RTX 4090 (24GB vRAM), 1TB RAM, 128 CPU cores, Debian (Bookworm), CUDA 12.5 and Python 3.11. As for LLMs, we decided on instruct models for our experiments such that they would run on our hardware and be comparable in size, leading to the following selection: Mistral (7B-Instruct) (Jiang et al., 2023), Llama2 (7B) (Touvron et al., 2023), Llama3 (8B-Instruct) (Dubey et al., 2024) and Qwen (2.5-7B-Instruct) (Yang et al., 2024) – for the rest of the paper we omit detailed version specifications.

4.2 Evaluation

We outperform related work on 4 out of 5 common datasets (AIDA (Yosef et al., 2011), KORE 50 (Hoffart et al., 2012), MSNBC (Cucerzan, 2007), ACE04 (Ratinov et al., 2011), AQUAINT (Milne and Witten, 2008)) in zero-shot settings despite our employed LLMs being at least an order of magnitude smaller[5]. We note that Ding et al. (2024b)

[5]EntGPT (Ding et al., 2024a) and ChatEL (Ding et al., 2024b) employ Llama2 70B (Touvron et al., 2023) and GPT-3.5 (Ouyang et al., 2022b); ChatEL (Ding et al., 2024b) additionally makes use of PaLM 540B (Chowdhery et al., 2023) and GPT-4 (OpenAI, 2023). OpenAI has not disclosed parameter counts for GPT-3.5 and GPT-4, but each of them is assumed to have at least 175B parameters, with rumors

argue that model parameter count has a significant influence on the entity disambiguation task.

We report our ED F1 results in Table 2. Our model performs strongly across all datasets and even surpasses finetuned or trained prior work in certain cases. Despite being a zero-shot approach, AGNUS attains overall new state-of-the-art results for KORE 50 (82.3%) and ACE04 (95.5%). Unsurprisingly when evaluating on AIDA, approaches trained on AIDA outperform ours, but AGNUS (86.7%) exceeds second-ranked zero-shot approach EntGPT-P (Ding et al., 2024a) (82.1%) F1 measure by 4.6%. Evaluating against KORE 50, AGNUS reaches 82.3% in comparison to ChatEL's 78.7%, surpassing it by 3.6%. As for ACE and AQUAINT, our results (95.5% and 87.5%) improve upon EntGPT-P's (91.8% and 79.1%) respectively by 3.7% and 8.4%. For MSNBC, we do not beat the state-of-the-art for zero-shot entity disambiguation and instead reach 82.4%, underperforming ChatEL (Ding et al., 2024b) (88.1%) by 5.7% and finetuned state-of-the-art CoherentED (96.3%) by 13.9%.

While AGNUS yields improvements across some benchmarks, we consider our primary benefit lying in enhancing disambiguation robustness via order invariance for candidates and by preventing structurally invalid outputs.

4.3 Ablation Study

AGNUS employs multiple techniques to mitigate issues relating to LLM-based ED. Particularly, AGNUS relies on LLMs for disambiguation, MACS for order-invariant candidate encoding, ACDC for entity decoding and particularly candidates' representation. In our ablation study, we therefore design experiments to verify the impact of these aspects on model results by investigating candidate representation (Section 4.3.1), LLM selection (Section 4.3.2), MACS (Section 4.3.3) and ACDC (Section 4.3.4).

4.3.1 Candidate Representation

To validate LLM disambiguation capabilities based on contextual candidate entity information, we apply AGNUS to candidate representations of different entity information types. We selected DBpedia (& Wikipedia) entity IRIs, entity types, textual entity descriptions and labels as meaningful entity information characterising entity candidates for our experiments (Tables 1 and 3). We note that in Table 1 across all datasets, IRI-based representations perform best with an average F1 performance of 86.9%, outperforming labels by 10.2% – with a tie of 87.9% for AQUAINT. For all datasets beside KORE 50 and AQUAINT, descriptions reach the second-highest score (avg.: 73.5%), but are still surpassed by labels (76.7%) on average by 3.2%. We note that the shorter and more unique a representation is, the better AGNUS seems to perform. In our experiments, we find effects of representation depend on benchmarked dataset with representation-based score differences ranging from 5.6% (ACE04) to 34.9% (AQUAINT) with a mean of 21.12% across our 6 datasets.

4.3.2 Large Language Model

To verify our approach's generalizability across LLMs, we run AGNUS on 4 LLMs: Llama2 (Touvron et al., 2023), Llama3 (Dubey et al., 2024), Qwen (Yang et al., 2024) and Mistral (Jiang et al., 2023). In Table 3, we notice similar trends across most LLMs for the AIDA dataset with Llama2 representing a slight outlier: All LLMs except for Llama2 attain their respective best results using IRIs (Qwen: 84.6%, Mistral: 86.7%, Llama3: 84.0%) as candidate information, whereas our outlier LLM manages to slightly improve on its 80.9% F1 measure, reaching 81.3% by employing labels as candidate representation. Typically, Llama3, Mistral and Qwen reach similar results to each other using IRIs (84.0% – 86.7%) and descriptions (70.2% – 76.3%) as candidate representations. Using labels, Qwen plummets down to 64.6%, whereas Llama3 (74.5%) and Mistral (74.3%) attain F1 scores close to each other. For type candidate information, Mistral (70.5%) noticeably outperforms Qwen (42.2%) and Llama3 (39.2%); Llama2 manages to outperform its successor Llama3 (56.0%). Llama3 (70.2%) and Qwen (72.1%) handle descriptions as meaningful entity information comparably well with Mistral (76.3%) performing slightly better and Llama2 (56.5%) displaying worst results.

4.3.3 Masked Attention Candidate Set

We investigate how MACS affects qualitative results and whether it actually renders disambiguation order-invariant. To this end, we run experiments shuffling candidates over 10 iterations and display results in Figure 4. Our experiments over 3 different LLMs (Llama3, Mistral, Qwen) dis-

claiming GPT-4 having 1.76 trillion parameters according to https://en.wikipedia.org/wiki/GPT-4.

Table 1: Ablation Study (Candidate Representation over datasets): AGNUS (Mistral) F1 measures on AIDA, AIDA-Syn, KORE 50, MSNBC, ACE04 and AQUAINT with different candidate entity representations (IRI, label, entity type, entity description), along with per representation and per dataset averages. Top entry by dataset in **bold**, second underlined.

Entity Representation	AIDA	AIDA-Syn	KORE 50	MSNBC	ACE04	AQUAINT	Mean
AGNUS w. IRI	**0.867**	**0.863**	**0.823**	**0.824**	**0.955**	**0.879**	0.869
AGNUS w. Label	0.743	0.706	0.785	0.589	0.899	**0.879**	0.767
AGNUS w. Type	0.705	0.719	0.595	0.591	0.934	0.530	0.679
AGNUS w. Description	0.763	0.790	0.515	0.679	0.954	0.706	0.735
Mean	0.769	0.770	0.679	0.671	0.936	0.748	0.762

Table 2: ED evaluation table – **Upper category**: ED systems trained on AIDA. **Lower category**: 0-shot ED systems (AGNUS , EntGPT-P, ChatEL). Top scores per column and category **bolded**, second highest underlined. Scores obtained from respective papers. Note that baseline with *hidden candidates* also uses matching to candidates (else naive results would tend to 0) and MACS ablations are run over multiple iterations, showing score variability.

Trained (or finetuned) on AIDA-CoNLL

Model	AIDA	KORE 50	MSNBC	ACE04	AQUAINT	Mean
End2End (Kolitsas et al., 2018)	0.891	0.569	0.933	0.892	0.894	0.836
GENRE (Cao et al., 2021)	0.933	0.542	0.943	0.901	0.899	0.844
REL (van Hulst et al., 2020)	0.928	0.618	0.935	0.897	0.873	0.850
ReFinED (Ayoola et al., 2022)	**0.939**	0.567	0.941	0.908	0.918	0.855
EntGPT-I (GPT3.5) (Ding et al., 2024a)	0.920	**0.753**	0.922	**0.937**	0.906	0.888
ExtEnD (Barba et al., 2022)	0.926	-	0.947	0.918	0.916	0.927
CoherentED (Xiao et al., 2023)	0.894	-	**0.963**	0.934	**0.946**	**0.934**

LLM 0-shot ED

Model	AIDA	KORE 50	MSNBC	ACE04	AQUAINT	Mean
ChatEL (Ding et al., 2024b)	-	0.787	**0.881**	0.893	0.767	0.832
EntGPT-P (GPT3.5) (Ding et al., 2024a)	0.821	0.716	0.867	0.918	0.791	0.823
EntGPT-P (Llama2 70B) (Ding et al., 2024a)	0.708	0.647	0.741	0.746	0.635	0.695
Ours – **AGNUS** (Llama2 8B)	0.809	0.529	0.562	0.897	0.576	0.675
Ours – **AGNUS** (Mistral)	**0.867**	**0.823**	0.824	**0.955**	**0.875**	**0.869**
Baseline: *Mistral (hidden candidates)*	0.791	0.794	0.739	0.953	0.720	0.799
Ablation: *w.o. MACS (best)*	0.865	0.811	0.814	0.962	0.907	(0.872)
Ablation: *w.o. MACS (worst)*	0.833	0.779	0.766	0.950	0.847	(0.835)

Table 3: Ablation Study (LLM, Candidate Representation, ACDC): AGNUS F1 measures for different types of candidate representations for Qwen, Mistral, Llama2, Llama3 and without constrained decoding via ACDC. AGNUS without ACDC utilises fuzzy search, ranking reply and candidate, matching to candidate with highest similarity.

Model	**AIDA**			
	IRI	Label	Type	Description
AGNUS (Qwen)	0.846	0.646	0.422	0.721
AGNUS (Mistral)	0.867	0.743	0.705	0.763
AGNUS (Llama2)	0.809	0.813	0.560	0.565
AGNUS (Llama3)	0.840	0.745	0.392	0.702
AGNUS (Llama3) *w.o. ACDC*	0.765	0.698	0.331	0.677

play how disambiguation varies without the use of MACS and remains unchanged when applying MACS. Order invariance persists across all 10 iterations of shuffled candidates when MACS is employed whereas not applying the causal mask to candidate entities yields result variations. Without MACS, Llama3 averages at 66.53% (MACS: 66.40%) and varies between 59.56% – 73.02%, a difference of 13.46%. Mistral on the other hand varies in the range of 32.47% – 43.69%, averaging at 38.07% without MACS across iterations of candidate shuffles (with MACS: 38.20%). In Table 2, we also display MACS ablations over 2 iterations, from one non-MACS execution to another exhibiting 3.7% F1 difference on average and beating AGNUS in ACE04 (96.2% vs. 95.5%). Finally, Qwen also exhibits changes resulting from candidate order changes: with an average of 65.57% (MACS: 64.54%) its candidate order-dependant results vary within the range 57.61% – 72.84%. Based on our experiments, we conclude that MACS effectively removes order-based bias from candidates with an overall minor average reduction in F1 score.

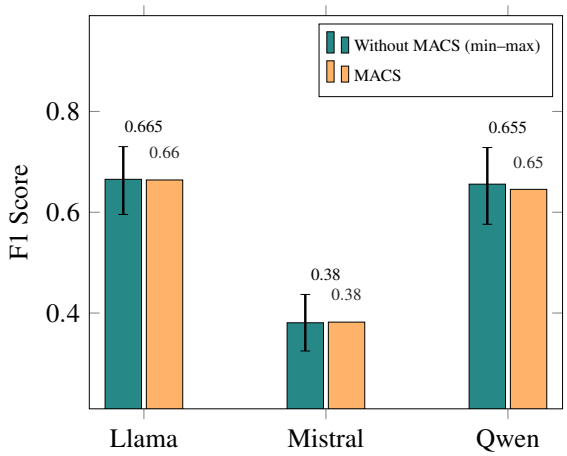

Figure 4: Ablation Study (MACS) - F1 Score Variability: Disambiguation without (*left*) and with MACS (*right*) with randomised candidate shuffles over 10 iterations with Llama3, Mistral and Qwen on perplexity decoding – disambiguates to highest confidence – for AIDA. MACS and non-MACS results are similar on average. Without MACS, performance varies (Llama3: 13.5%, Mistral: 11.2%, Qwen: 15.2%).

4.3.4 Agnus Contextual Decoding

In Table 3, additionally to checking out the impact of candidate representations across language models, we also evaluate AGNUS without our constrained decoding method (ACDC). With this setup, we find that AGNUS hallucinates across the board, decreasing F1 scores for all types of candidate representation. In non-ACDC experiments, we apply fuzzy matching to improve the likelihood of finding at least one entity. Exact disambiguation to candidate matches in our zero-shot experiments yield extremely subpar results (close to 0) and would otherwise be misrepresenting (exaggerating) the added value of our robustness-oriented approach. On average, F1 performance without ACDC is lowered by 5.2%, the largest drops appearing with IRI (-7.5%) and type (-6.1%) candidate representations, followed by label (-4.7%) and descriptions (-2.5%).

4.4 Contamination Detection & Mitigation

To estimate potential contamination, we employ perplexity (Li, 2023) to quantify a model's uncertainty for a given token sequence prediction. Perplexity reflects the inverse likelihood assigned to a particular token sequence by a model: lower perplexity indicates higher predictive confidence and a higher likelihood of contamination. To mitigate contamination and evaluate the generalizability of LLMs, we propose synthetically generating a novel dataset derived from an existing one by replacing each entity mention with a distinct, contextually similar mention and corresponding entity. We apply our method with the DeepSeek-R1 (DeepSeek-AI et al., 2024) model[6] to AIDA (Yosef et al., 2011) and release AIDA-Syn[7]. For each sequence, we produced five alternative mention-entity sets, but for AIDA-Syn only one was retained per instance to reduce the risk of future pretraining exposure. All alternatives, along with a generation script, are made available[8]. To assess contamination levels across different LLMs, we introduce a modified decoding strategy, illustrated in Figure 4 with disambiguation performed by selecting a candidate entity with highest confidence. The model that performs worst with this strategy is presumed to be least contaminated. Our findings show that Mistral (Jiang et al., 2023) yields the lowest performance with a perplexity-based decoding method on AIDA, suggesting being least affected by benchmark contamination. Applying the same decoding strategy with AIDA-Syn, F1 score decreases from 38.20% to 22.82%, a substantial relative drop of 15.38%. This reduction supports the hypothesis that AIDA-Syn exhibits reduced contamination.

These results demonstrate that our approach may effectively mitigate benchmark contamination and provides a more robust basis for evaluating the generalization capabilities of LLMs for ED.

5 Conclusion

We propose a set of techniques to enable robust LLM-based entity disambiguation by addressing the issues of unwanted order bias and hallucinations.Our experimental results show that by doing so, our zero-shot approach outperforms prior work on average by 3.5%. Further, we introduce a methodology to mitigate evaluation contamination and publish a novel dataset AIDA-Syn based on AIDA, along with code to generate different versions of it. While our approach using MACS and ACDC yields modest average improvements across benchmarks, our primary benefit lies in enhancing output robustness and controlling generation behavior, particularly in cases where unconstrained and order-variant decoding leads to semantically or structurally invalid outputs.

[6]Version from May 2025: https://www.deepseek.com/

[7]The existence and validity of all entities were verified using the DBpedia SPARQL endpoint, resulting in a final dataset of 888 documents.

[8]https://anonymous.4open.science/r/Agnus/README.md

6 Limitations

Due to our introduction of order-invariance by application of a causal mask and modifying positional embeddings, we are limited to open source LLMs, making evaluation with DeepSeek (DeepSeek-AI et al., 2024), GPT-3.5 (Ouyang et al., 2022b), GPT-4 (OpenAI, 2023) impossible.

Alike other deep learning approaches to entity disambiguation, AGNUS is limited by its generated candidate sets and by only working with candidate entities that have some form of textual label, description, types or otherwise meaningful information for a LLM to predict.

While ACDC does mitigate hallucinations, a given LLM's next token prediction may be to continue with non-entity tokens, such as a greeting or similar, therewith potentially negatively affecting entity disambiguation. Designing a specific decoding strategy to include such behaviour could be a potential benefit in the future.

In this paper, our models are not finetuned for the entity disambiguation task nor given particular domain-specific information that could boost their information. Therefore, we concede that going for a few-shot approach could yield improved results.

Further, despite having the out-of-the-box structural capabilities for it, we could not evaluate our approach on knowledge bases other than Wikipedia or DBpedia due to not being aware of comparable and valid evaluation benchmarks for it.

LLMs are language-dependant and have mainly been trained with English in mind. We only benchmark our system

Regarding evaluation contamination and the creation of `AIDA-Syn`, we did not go as in-depth explaining our procedure, safeguards against LLM hallucinations, inherent surrounding bias as we would have liked, nor provide in-depth statistics or analyses. We use it mainly to evaluate our approach and show that despite there being novel entities and candidates, AGNUS is capable of attaining similar results as for the non-synthetic version with the suggested least contaminated LLM.

References

Anonymous. 2025. Removed to honour ACL anonymity policies. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (ACL)*. Accepted for publication.

Tom Ayoola, Shubhi Tyagi, Joseph Fisher, Christos Christodoulopoulos, and Andrea Pierleoni. 2022. ReFined: An efficient zero-shot-capable approach to end-to-end entity linking. In *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies: Industry Track, NAACL 2022, Hybrid: Seattle, Washington, USA + Online, July 10-15, 2022*, pages 209–220. Association for Computational Linguistics.

Edoardo Barba, Luigi Procopio, and Roberto Navigli. 2022. ExtEnD: Extractive entity disambiguation. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics*, Online and Dublin, Ireland. Association for Computational Linguistics.

Emily M. Bender, Timnit Gebru, Angelina McMillan-Major, and Shmargaret Shmitchell. 2021. On the dangers of stochastic parrots: Can language models be too big? In *FAccT '21: 2021 ACM Conference on Fairness, Accountability, and Transparency, Virtual Event / Toronto, Canada, March 3-10, 2021*, pages 610–623. ACM.

Luca Beurer-Kellner, Marc Fischer, and Martin T. Vechev. 2024. Guiding llms the right way: Fast, non-invasive constrained generation. In *Forty-first International Conference on Machine Learning, ICML 2024, Vienna, Austria, July 21-27, 2024*. OpenReview.net.

Tom B. Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel M. Ziegler, Jeffrey Wu, Clemens Winter, Christopher Hesse, Mark Chen, Eric Sigler, Mateusz Litwin, Scott Gray, Benjamin Chess, Jack Clark, Christopher Berner, Sam McCandlish, Alec Radford, Ilya Sutskever, and Dario Amodei. 2020. Language models are few-shot learners. In *Advances in Neural Information Processing Systems 33: Annual Conference on Neural Information Processing Systems 2020, NeurIPS 2020, December 6-12, 2020, virtual*.

Nicola De Cao, Gautier Izacard, Sebastian Riedel, and Fabio Petroni. 2021. Autoregressive entity retrieval. In *9th International Conference on Learning Representations, ICLR 2021, Virtual Event, Austria, May 3-7, 2021*. OpenReview.net.

Simin Chen, Yiming Chen, Zexin Li, Yifan Jiang, Zhongwei Wan, Yixin He, Dezhi Ran, Tianle Gu, Haizhou Li, Tao Xie, and Baishakhi Ray. 2025. Recent advances in large langauge model benchmarks against data contamination: From static to dynamic evaluation. *CoRR*, abs/2502.17521.

Aakanksha Chowdhery, Sharan Narang, Jacob Devlin, Maarten Bosma, Gaurav Mishra, Adam Roberts, Paul Barham, Hyung Won Chung, Charles Sutton, Sebastian Gehrmann, Parker Schuh, Kensen Shi, Sasha Tsvyashchenko, Joshua Maynez, Abhishek

Rao, Parker Barnes, Yi Tay, Noam Shazeer, Vinodkumar Prabhakaran, Emily Reif, Nan Du, Ben Hutchinson, Reiner Pope, James Bradbury, Jacob Austin, Michael Isard, Guy Gur-Ari, Pengcheng Yin, Toju Duke, Anselm Levskaya, Sanjay Ghemawat, Sunipa Dev, Henryk Michalewski, Xavier Garcia, Vedant Misra, Kevin Robinson, Liam Fedus, Denny Zhou, Daphne Ippolito, David Luan, Hyeontaek Lim, Barret Zoph, Alexander Spiridonov, Ryan Sepassi, David Dohan, Shivani Agrawal, Mark Omernick, Andrew M. Dai, Thanumalayan Sankaranarayana Pillai, Marie Pellat, Aitor Lewkowycz, Erica Moreira, Rewon Child, Oleksandr Polozov, Katherine Lee, Zongwei Zhou, Xuezhi Wang, Brennan Saeta, Mark Diaz, Orhan Firat, Michele Catasta, Jason Wei, Kathy Meier-Hellstern, Douglas Eck, Jeff Dean, Slav Petrov, and Noah Fiedel. 2023. Palm: Scaling language modeling with pathways. *J. Mach. Learn. Res.*, 24:240:1–240:113.

Silviu Cucerzan. 2007. Large-scale named entity disambiguation based on wikipedia data. In *EMNLP-CoNLL 2007, Proceedings of the 2007 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning, June 28-30, 2007, Prague, Czech Republic*, pages 708–716. ACL.

DeepSeek-AI, Aixin Liu, Bei Feng, Bing Xue, Bingxuan Wang, Bochao Wu, Chengda Lu, Chenggang Zhao, Chengqi Deng, Chenyu Zhang, Chong Ruan, Damai Dai, Daya Guo, Dejian Yang, Deli Chen, Dongjie Ji, Erhang Li, Fangyun Lin, Fucong Dai, Fuli Luo, Guangbo Hao, Guanting Chen, Guowei Li, H. Zhang, Han Bao, Hanwei Xu, Haocheng Wang, Haowei Zhang, Honghui Ding, Huajian Xin, Huazuo Gao, Hui Li, Hui Qu, J. L. Cai, Jian Liang, Jianzhong Guo, Jiaqi Ni, Jiashi Li, Jiawei Wang, Jin Chen, Jingchang Chen, Jingyang Yuan, Junjie Qiu, Junlong Li, Junxiao Song, Kai Dong, Kai Hu, Kaige Gao, Kang Guan, Kexin Huang, Kuai Yu, Lean Wang, Lecong Zhang, Lei Xu, Leyi Xia, Liang Zhao, Litong Wang, Liyue Zhang, Meng Li, Miaojun Wang, Mingchuan Zhang, Minghua Zhang, Minghui Tang, Mingming Li, Ning Tian, Panpan Huang, Peiyi Wang, Peng Zhang, Qiancheng Wang, Qihao Zhu, Qinyu Chen, Qiushi Du, R. J. Chen, R. L. Jin, Ruiqi Ge, Ruisong Zhang, Ruizhe Pan, Runji Wang, Runxin Xu, Ruoyu Zhang, Ruyi Chen, S. S. Li, Shanghao Lu, Shangyan Zhou, Shanhuang Chen, Shaoqing Wu, Shengfeng Ye, Shengfeng Ye, Shirong Ma, Shiyu Wang, Shuang Zhou, Shuiping Yu, Shunfeng Zhou, Shuting Pan, T. Wang, Tao Yun, Tian Pei, Tianyu Sun, W. L. Xiao, and Wangding Zeng. 2024. Deepseek-v3 technical report. *CoRR*, abs/2412.19437.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL-HLT 2019, Minneapolis, MN, USA, June 2-7, 2019, Volume 1 (Long and Short Papers)*, pages 4171–4186. Association for Computational Linguistics.

Yifan Ding, Amrit Poudel, Qingkai Zeng, Tim Weninger, Balaji Veeramani, and Sanmitra Bhattacharya. 2024a. Entgpt: Linking generative large language models with knowledge bases. *CoRR*, abs/2402.06738.

Yifan Ding, Qingkai Zeng, and Tim Weninger. 2024b. Chatel: Entity linking with chatbots. In *Proceedings of the 2024 Joint International Conference on Computational Linguistics, Language Resources and Evaluation, LREC/COLING 2024, 20-25 May, 2024, Torino, Italy*, pages 3086–3097. ELRA and ICCL.

Ricardo Dominguez-Olmedo, Moritz Hardt, and Celestine Mendler-Dünner. 2023. Questioning the survey responses of large language models. *arXiv preprint arXiv:2306.07951*.

Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Amy Yang, Angela Fan, Anirudh Goyal, Anthony Hartshorn, Aobo Yang, Archi Mitra, Archie Sravankumar, Artem Korenev, Arthur Hinsvark, Arun Rao, Aston Zhang, Aurélien Rodriguez, Austen Gregerson, Ava Spataru, Baptiste Rozière, Bethany Biron, Binh Tang, Bobbie Chern, Charlotte Caucheteux, Chaya Nayak, Chloe Bi, Chris Marra, Chris McConnell, Christian Keller, Christophe Touret, Chunyang Wu, Corinne Wong, Cristian Canton Ferrer, Cyrus Nikolaidis, Damien Allonsius, Daniel Song, Danielle Pintz, Danny Livshits, David Esiobu, Dhruv Choudhary, Dhruv Mahajan, Diego Garcia-Olano, Diego Perino, Dieuwke Hupkes, Egor Lakomkin, Ehab AlBadawy, Elina Lobanova, Emily Dinan, Eric Michael Smith, Filip Radenovic, Frank Zhang, Gabriel Synnaeve, Gabrielle Lee, Georgia Lewis Anderson, Graeme Nail, Grégoire Mialon, Guan Pang, Guillem Cucurell, Hailey Nguyen, Hannah Korevaar, Hu Xu, Hugo Touvron, Iliyan Zarov, Imanol Arrieta Ibarra, Isabel M. Kloumann, Ishan Misra, Ivan Evtimov, Jade Copet, Jaewon Lee, Jan Geffert, Jana Vranes, Jason Park, Jay Mahadeokar, Jeet Shah, Jelmer van der Linde, Jennifer Billock, Jenny Hong, Jenya Lee, Jeremy Fu, Jianfeng Chi, Jianyu Huang, Jiawen Liu, Jie Wang, Jiecao Yu, Joanna Bitton, Joe Spisak, Jongsoo Park, Joseph Rocca, Joshua Johnstun, Joshua Saxe, Junteng Jia, Kalyan Vasuden Alwala, Kartikeya Upasani, Kate Plawiak, Ke Li, Kenneth Heafield, Kevin Stone, and et al. 2024. The llama 3 herd of models. *CoRR*, abs/2407.21783.

Johannes Hoffart, Stephan Seufert, Dat Ba Nguyen, Martin Theobald, and Gerhard Weikum. 2012. KORE: keyphrase overlap relatedness for entity disambiguation. In *21st ACM International Conference on Information and Knowledge Management, CIKM'12, Maui, HI, USA, October 29 - November 02, 2012*, pages 545–554. ACM.

Ari Holtzman, Jan Buys, Li Du, Maxwell Forbes, and Yejin Choi. 2020. The curious case of neural text

degeneration. In *8th International Conference on Learning Representations, ICLR 2020, Addis Ababa, Ethiopia, April 26-30, 2020*. OpenReview.net.

Lei Huang, Weijiang Yu, Weitao Ma, Weihong Zhong, Zhangyin Feng, Haotian Wang, Qianglong Chen, Weihua Peng, Xiaocheng Feng, Bing Qin, and Ting Liu. 2025. A survey on hallucination in large language models: Principles, taxonomy, challenges, and open questions. *ACM Transactions on Information Systems*, 43(2):1–55.

Ziwei Ji, Nayeon Lee, Rita Frieske, Tiezheng Yu, Dan Su, Yan Xu, Etsuko Ishii, Yejin Bang, Andrea Madotto, and Pascale Fung. 2023. Survey of hallucination in natural language generation. *ACM Comput. Surv.*, 55(12):248:1–248:38.

Albert Q. Jiang, Alexandre Sablayrolles, Arthur Mensch, Chris Bamford, Devendra Singh Chaplot, Diego de Las Casas, Florian Bressand, Gianna Lengyel, Guillaume Lample, Lucile Saulnier, Lélio Renard Lavaud, Marie-Anne Lachaux, Pierre Stock, Teven Le Scao, Thibaut Lavril, Thomas Wang, Timothée Lacroix, and William El Sayed. 2023. Mistral 7b. *CoRR*, abs/2310.06825.

Nikolaos Kolitsas, Octavian-Eugen Ganea, and Thomas Hofmann. 2018. End-to-end neural entity linking. In *Proceedings of the 22nd Conference on Computational Natural Language Learning, CoNLL 2018, Brussels, Belgium, October 31 - November 1, 2018*, pages 519–529. Association for Computational Linguistics.

Patrick Lewis, Ethan Perez, Aleksandra Piktus, Fabio Petroni, Vladimir Karpukhin, Naman Goyal, Heinrich Küttler, Mike Lewis, Wen-tau Yih, Tim Rocktäschel, Sebastian Riedel, and Douwe Kiela. 2020. Retrieval-augmented generation for knowledge-intensive NLP tasks. In *Advances in Neural Information Processing Systems 33: Annual Conference on Neural Information Processing Systems 2020, NeurIPS 2020, December 6-12, 2020, virtual*.

Ruizhe Li and Yanjun Gao. 2024. Anchored answers: Unravelling positional bias in gpt-2's multiple-choice questions. *arXiv preprint arXiv:2405.03205*.

Yucheng Li. 2023. Estimating contamination via perplexity: Quantifying memorisation in language model evaluation. *CoRR*, abs/2309.10677.

Zongjie Li, Chaozheng Wang, Pingchuan Ma, Daoyuan Wu, Shuai Wang, Cuiyun Gao, and Yang Liu. 2023. Split and merge: Aligning position biases in large language model based evaluators. *arXiv preprint arXiv:2310.01432*.

Aman Madaan, Niket Tandon, Prakhar Gupta, Skyler Hallinan, Luyu Gao, Sarah Wiegreffe, Uri Alon, Nouha Dziri, Shrimai Prabhumoye, Yiming Yang, Shashank Gupta, Bodhisattwa Prasad Majumder, Katherine Hermann, Sean Welleck, Amir Yazdanbakhsh, and Peter Clark. 2023. Self-refine: Iterative refinement with self-feedback. In *Advances in Neural Information Processing Systems 36: Annual Conference on Neural Information Processing Systems 2023, NeurIPS 2023, New Orleans, LA, USA, December 10 - 16, 2023*.

Joshua Maynez, Shashi Narayan, Bernd Bohnet, and Ryan T. McDonald. 2020. On faithfulness and factuality in abstractive summarization. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, ACL 2020, Online, July 5-10, 2020*, pages 1906–1919. Association for Computational Linguistics.

David N. Milne and Ian H. Witten. 2008. Learning to link with wikipedia. In *Proceedings of the 17th ACM Conference on Information and Knowledge Management, CIKM 2008, Napa Valley, California, USA, October 26-30, 2008*, pages 509–518. ACM.

Andrea Moro, Alessandro Raganato, and Roberto Navigli. 2014. Entity Linking meets Word Sense Disambiguation: a Unified Approach. *Transactions of the Association for Computational Linguistics (TACL)*, 2:231–244.

OpenAI. 2023. GPT-4 technical report. *CoRR*, abs/2303.08774.

Riccardo Orlando, Pere-Lluís Huguet Cabot, Edoardo Barba, and Roberto Navigli. 2024. Retrieve, read and link: Fast and accurate entity linking and relation extraction on an academic budget. In *Findings of the Association for Computational Linguistics: ACL 2024*, Bangkok, Thailand. Association for Computational Linguistics.

Long Ouyang, Jeffrey Wu, Xu Jiang, Diogo Almeida, Carroll L. Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, John Schulman, Jacob Hilton, Fraser Kelton, Luke Miller, Maddie Simens, Amanda Askell, Peter Welinder, Paul F. Christiano, Jan Leike, and Ryan Lowe. 2022a. Training language models to follow instructions with human feedback. In *Advances in Neural Information Processing Systems 35: Annual Conference on Neural Information Processing Systems 2022, NeurIPS 2022, New Orleans, LA, USA, November 28 - December 9, 2022*.

Long Ouyang, Jeffrey Wu, Xu Jiang, Diogo Almeida, Carroll L. Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, John Schulman, Jacob Hilton, Fraser Kelton, Luke Miller, Maddie Simens, Amanda Askell, Peter Welinder, Paul F. Christiano, Jan Leike, and Ryan Lowe. 2022b. Training language models to follow instructions with human feedback. In *Advances in Neural Information Processing Systems 35: Annual Conference on Neural Information Processing Systems 2022, NeurIPS 2022, New Orleans, LA, USA, November 28 - December 9, 2022*.

Fabio Petroni, Aleksandra Piktus, Angela Fan, Patrick Lewis, Majid Yazdani, Nicola De Cao, James Thorne, Yacine Jernite, Vladimir Karpukhin, Jean Maillard,

Vassilis Plachouras, Tim Rocktäschel, and Sebastian Riedel. 2021. KILT: a benchmark for knowledge intensive language tasks. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL-HLT 2021, Online, June 6-11, 2021*, pages 2523–2544. Association for Computational Linguistics.

Pouya Pezeshkpour and Estevam Hruschka. 2023. Large language models sensitivity to the order of options in multiple-choice questions. *arXiv preprint arXiv:2308.11483*.

Francesco Piccinno and Paolo Ferragina. 2014. From tagme to WAT: a new entity annotator. In *ERD'14, Proceedings of the First ACM International Workshop on Entity Recognition & Disambiguation, July 11, 2014, Gold Coast, Queensland, Australia*, pages 55–62. ACM.

Larissa Pusch and Tim O. F. Conrad. 2024. Combining llms and knowledge graphs to reduce hallucinations in question answering. *CoRR*, abs/2409.04181.

Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, Ilya Sutskever, et al. 2019. Language models are unsupervised multitask learners. *OpenAI blog*, 1(8):9.

Jonathan Raiman and Olivier Raiman. 2018. Deeptype: Multilingual entity linking by neural type system evolution. In *Proceedings of the Thirty-Second AAAI Conference on Artificial Intelligence, (AAAI-18), the 30th innovative Applications of Artificial Intelligence (IAAI-18), and the 8th AAAI Symposium on Educational Advances in Artificial Intelligence (EAAI-18), New Orleans, Louisiana, USA, February 2-7, 2018*, pages 5406–5413. AAAI Press.

Lev-Arie Ratinov, Dan Roth, Doug Downey, and Mike Anderson. 2011. Local and global algorithms for disambiguation to wikipedia. In *The 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies, Proceedings of the Conference, 19-24 June, 2011, Portland, Oregon, USA*, pages 1375–1384. The Association for Computer Linguistics.

Pranab Sahoo, Ayush Kumar Singh, Sriparna Saha, Vinija Jain, Samrat Mondal, and Aman Chadha. 2024. A systematic survey of prompt engineering in large language models: Techniques and applications. *CoRR*, abs/2402.07927.

Hassan Shavarani and Anoop Sarkar. 2023. Spel: Structured prediction for entity linking. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing, EMNLP 2023, Singapore, December 6-10, 2023*, pages 11123–11137. Association for Computational Linguistics.

Hugo Touvron, Louis Martin, Kevin Stone, Peter Albert, Amjad Almahairi, Yasmine Babaei, Nikolay Bashlykov, Soumya Batra, Prajjwal Bhargava, Shruti Bhosale, Dan Bikel, Lukas Blecher, Cristian Canton-Ferrer, Moya Chen, Guillem Cucurull, David Esiobu, Jude Fernandes, Jeremy Fu, Wenyin Fu, Brian Fuller, Cynthia Gao, Vedanuj Goswami, Naman Goyal, Anthony Hartshorn, Saghar Hosseini, Rui Hou, Hakan Inan, Marcin Kardas, Viktor Kerkez, Madian Khabsa, Isabel Kloumann, Artem Korenev, Punit Singh Koura, Marie-Anne Lachaux, Thibaut Lavril, Jenya Lee, Diana Liskovich, Yinghai Lu, Yuning Mao, Xavier Martinet, Todor Mihaylov, Pushkar Mishra, Igor Molybog, Yixin Nie, Andrew Poulton, Jeremy Reizenstein, Rashi Rungta, Kalyan Saladi, Alan Schelten, Ruan Silva, Eric Michael Smith, Ranjan Subramanian, Xiaoqing Ellen Tan, Binh Tang, Ross Taylor, Adina Williams, Jian Xiang Kuan, Puxin Xu, Zheng Yan, Iliyan Zarov, Yuchen Zhang, Angela Fan, Melanie Kambadur, Sharan Narang, Aurélien Rodriguez, Robert Stojnic, Sergey Edunov, and Thomas Scialom. 2023. Llama 2: Open foundation and fine-tuned chat models. *CoRR*, abs/2307.09288.

Johannes M. van Hulst, Faegheh Hasibi, Koen Dercksen, Krisztian Balog, and Arjen P. de Vries. 2020. Rel: An entity linker standing on the shoulders of giants. In *Proceedings of the 43rd International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '20, page 2197–2200, New York, NY, USA. Association for Computing Machinery.

Ruben Verborgh, Michael Röder, Ricardo Usbeck, and Axel-Cyrille Ngonga Ngomo. 2018. Gerbil – benchmarking named entity recognition and linking consistently. *Semant. Web*, 9(5):605–625.

Peiyi Wang, Lei Li, Liang Chen, Zefan Cai, Dawei Zhu, Binghuai Lin, Yunbo Cao, Qi Liu, Tianyu Liu, and Zhifang Sui. 2023. Large language models are not fair evaluators. *arXiv preprint arXiv:2305.17926*.

Siyuan Wang, Zhuohan Long, Zhihao Fan, Xuanjing Huang, and Zhongyu Wei. 2025. Benchmark self-evolving: A multi-agent framework for dynamic LLM evaluation. In *Proceedings of the 31st International Conference on Computational Linguistics, COLING 2025, Abu Dhabi, UAE, January 19-24, 2025*, pages 3310–3328. Association for Computational Linguistics.

Xinpeng Wang, Chengzhi Hu, Bolei Ma, Paul Röttger, and Barbara Plank. 2024a. Look at the text: Instruction-tuned language models are more robust multiple choice selectors than you think. *arXiv preprint arXiv:2404.08382*.

Xinpeng Wang, Bolei Ma, Chengzhi Hu, Leon Weber-Genzel, Paul Röttger, Frauke Kreuter, Dirk Hovy, and Barbara Plank. 2024b. "my answer is c": First-token probabilities do not match text answers in instruction-tuned language models. *arXiv preprint arXiv:2402.14499*.

Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Brian Ichter, Fei Xia, Ed H. Chi, Quoc V. Le, and Denny Zhou. 2022. Chain-of-thought prompting

elicits reasoning in large language models. In *Advances in Neural Information Processing Systems 35: Annual Conference on Neural Information Processing Systems 2022, NeurIPS 2022, New Orleans, LA, USA, November 28 - December 9, 2022*.

Sheng-Lun Wei, Cheng-Kuang Wu, Hen-Hsen Huang, and Hsin-Hsi Chen. 2024. Unveiling selection biases: Exploring order and token sensitivity in large language models. *arXiv preprint arXiv:2406.03009*.

Ledell Wu, Fabio Petroni, Martin Josifoski, Sebastian Riedel, and Luke Zettlemoyer. 2020. Scalable zero-shot entity linking with dense entity retrieval. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing, EMNLP 2020, Online, November 16-20, 2020*, pages 6397–6407. Association for Computational Linguistics.

Zilin Xiao, Linjun Shou, Xingyao Zhang, Jie Wu, Ming Gong, and Daxin Jiang. 2023. Coherent entity disambiguation via modeling topic and categorical dependency. In *Findings of the Association for Computational Linguistics: EMNLP 2023, Singapore, December 6-10, 2023*, pages 7480–7492. Association for Computational Linguistics.

Cheng Xu, Shuhao Guan, Derek Greene, and M. Tahar Kechadi. 2024. Benchmark data contamination of large language models: A survey. *CoRR*, abs/2406.04244.

Mengge Xue, Zhenyu Hu, Liqun Liu, Kuo Liao, Shuang Li, Honglin Han, Meng Zhao, and Chengguo Yin. 2024. Strengthened symbol binding makes large language models reliable multiple-choice selectors. *arXiv preprint arXiv:2406.01026*.

An Yang, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chengyuan Li, Dayiheng Liu, Fei Huang, Haoran Wei, Huan Lin, Jian Yang, Jianhong Tu, Jianwei Zhang, Jianxin Yang, Jiaxi Yang, Jingren Zhou, Junyang Lin, Kai Dang, Keming Lu, Keqin Bao, Kexin Yang, Le Yu, Mei Li, Mingfeng Xue, Pei Zhang, Qin Zhu, Rui Men, Runji Lin, Tianhao Li, Tingyu Xia, Xingzhang Ren, Xuancheng Ren, Yang Fan, Yang Su, Yichang Zhang, Yu Wan, Yuqiong Liu, Zeyu Cui, Zhenru Zhang, and Zihan Qiu. 2024. Qwen2.5 technical report. *CoRR*, abs/2412.15115.

Mohamed Amir Yosef, Johannes Hoffart, Ilaria Bordino, Marc Spaniol, and Gerhard Weikum. 2011. AIDA: an online tool for accurate disambiguation of named entities in text and tables. *Proc. VLDB Endow.*, 4(12):1450–1453.

Zihao Zhao, Eric Wallace, Shi Feng, Dan Klein, and Sameer Singh. 2021. Calibrate before use: Improving few-shot performance of language models. In *International conference on machine learning*, pages 12697–12706. PMLR.

Chujie Zheng, Hao Zhou, Fandong Meng, Jie Zhou, and Minlie Huang. 2023. Large language models are not robust multiple choice selectors. In *The Twelfth International Conference on Learning Representations*.

Kaijie Zhu, Jiaao Chen, Jindong Wang, Neil Zhenqiang Gong, Diyi Yang, and Xing Xie. 2024a. Dyval: Dynamic evaluation of large language models for reasoning tasks. In *The Twelfth International Conference on Learning Representations, ICLR 2024, Vienna, Austria, May 7-11, 2024*. OpenReview.net.

Kaijie Zhu, Jindong Wang, Qinlin Zhao, Ruochen Xu, and Xing Xie. 2024b. Dyval 2: Dynamic evaluation of large language models by meta probing agents. *CoRR*, abs/2402.14865.

A Appendix

Over the course of researching and developing AGNUS, we implemented some further aspects that we could not allude to in depth. Here are some supplemental materials about them that might be of interest to fellow researchers.

A.1 AIDA-Syn

We created AIDA-Syn using DeepSeek-R1 and generated 5 variants of coherent mentions and entities each. We filtered out variants and documents where entities did not correspond to a valid DBpedia entity or where other LLM-related issues may have arisen. Some issues were related to DeepSeek's maximum number of generated tokens activating prior to reaching the end. Key criteria for our generation included semantic coherence, lexical diversity, naturalness, plausibility within the surrounding text, and alignment with existing entities – our employed setups and prompts are publicly available[9]. In the end, we generated a collection of 888 synthetic documents with multiple variants of mentions and entities for each. We would have liked to employ ACDC for DeepSeek's suggested entities, but unfortunately the API does not allow for finegranular control and our hardware limitations did not allow for us to run the high-parameter and high-performing models.

As a means of verifying that our generated mentions and entities are sensical, we used a two-pronged approach. First, one researcher manually went through a random sample of 100 documents, verifying contextual coherence for all variants. Second, we attempted to run the full suite of annotators via GERBIL (Verborgh et al., 2018) to see whether existing approaches could annotate documents effectively – we report the results in Table 6. Unfortunately, many D2KB annotators did not run on our full `AIDA-Syn` (or ASM-10[10], ASM-50[11], ASM-100[12]) and the original `AIDA` datasets, instead returning timeout errors and similar. In Table 5 we display some details about the synthetic `AIDA-Syn` dataset including number of documents (888), total number of mentions (15,314) as well as entity type consistency between the original dataset and the transformed documents. Our assumption is that a certain degree of overlap between types should persist, but that it shouldn't be an absolute overlap the sake of document diversity.

Due to licensing, our datasets are provided as annotations without contextual input document text, but with our own novel mentions, entities and offsets. Unfortunately, researchers may want to look for contextual clues to inject our annotations into relevant documents. Hypothetically, such may even be possible by potentially matching with a source NIF-processed data (e.g. GERBIL) – which we cannot recommend with a clear conscience and would not condone nor support by any means at our disposal whether through code[13] or otherwise.

A.2 Context Length

The number of tokens an LLM may process at once and is therefore limited to is known as *context length*. If tokens surpass the maximal context length an LLM was trained for, it produces gibberish at an increased likelihood – to the point that some LLMs will opt to instead raise an error when a threshold is reached. To counteract the issue of context length for LLM-based entity disambiguation, we introduce *Hierarchical Elimination Tree Disambiguation*, a linearly scalable disambiguation method for iterative pruning of unwanted candidate entities – alike single-elimination tournaments.

We did not run into issues relating to context length in our experiments when comparing with prior work due to the limited number of candidates. Regardless, we developed a relatively simple approach allowing to sidestep the context length issue (see Figure 5).

The approach resolves the problem of context length by transforming the disambiguation task of $|C|$ candidates into tasks of smaller subsets of at most k disambiguation candidates instead, aggregating results and repeating the AGNUS disambiguation process (see Fig. 1) with further subsets of candidate entities until disambiguation converges on one entity. Formally, with $N = |C|$ entity candidates, k maximum threshold for concurrent candidates and $j \in (1, .., \lceil \log_k N \rceil)$, Hierarchical Elimination Tree Disambiguation (HET) leads to $1 + \sum_{j=1}^{\lceil log_k(n) \rceil} \lceil \frac{n}{k^j} \rceil$ disambiguation tasks of at most size k each being computed.

[9] https://anonymous.4open.science/r/Agnus/README.md

[10] http://gerbil.aksw.org/gerbil/experiment?id=202505190000

[11] http://gerbil.aksw.org/gerbil/experiment?id=202505190001

[12] http://gerbil.aksw.org/gerbil/experiment?id=202505190002

[13] https://anonymous.4open.science/r/Agnus/README.md

Table 4: Ablation Study (Entity Representation - Single and Pairwise): Disambiguation results (F1-measure) on AIDA for pairwise and singular (diagonal) entity representation information types for candidates on AGNUS (Mistral): entity IRI, entity type(s), entity label and entity description. Per column top-ranked score in **bold**, second-ranked underlined.

Entity Representation	AGNUS w. IRI	AGNUS w. Type	AGNUS w. Label	AGNUS w. Desc.
AGNUS w. IRI	**0.867**	**0.855**	**0.763**	**0.854**
AGNUS w. Type	0.855	0.705	0.734	0.766
AGNUS w. Label	0.763	0.734	0.743	0.744
AGNUS w. Desc.	0.854	0.766	0.744	0.763

Table 5: Some data statistics for AIDA-Syn. Type-Consistency compares pre-transformation types of entities to post-transformation types of entities and checks overlap.

# Documents	Mentions	# Type-Consistent Docs.	Type Consist. (Mean)
888	15,314	331	46.60%

System	AIDA-Syn	AIDA	ASM-10	ASM-50	ASM-100
Babelfy (Moro et al., 2014)	0.7503	0.6729	0.7660	0.7111	0.6912
WAT (Piccinno and Ferragina, 2014)	0.8641	0.6986	0.9355	0.8235	0.8332
REL (van Hulst et al., 2020)	-	?	0.9030	0.7942	0.6829

Table 6: F1 measures on datasets AIDA-Syn, AIDA for AGNUS and GERBIL-available systems (all other publicly available systems on GERBIL (Verborgh et al., 2018) timed out or returned "*The annotator caused too many single errors*" for the platform despite repeated attempts).

AIDA*	100 cands., $k = 2$	10 candidates
AGNUS (Mistral)	0.8329	0.8669

Table 7: ED for AGNUS on 581 AIDA documents with 100 candidates (HET, $k = 2$) and 10 candidates.

In Table 7, we show the result of a computed 'stress test' for HET with 100 candidates and $k = 2$ to maximise the number of disambiguation runs to see how much performance would deteriorate for AIDA. Our 'worst-case' HET experiment creates 7 elimination rounds and 98 disambiguation tasks for each mention. By doing so, the likelihood of potentially propagating errors increases, but the performance difference between our HET-activated stress test and usual-setting AGNUS is only 3.4% for AIDA[14]. Nevertheless, we do not recommend running HET with $k = 2$ unless absolutely necessary for a small context length (e.g. when introducing multiple shots) due to the unnecessarily large amount of disambiguations to be performed for large candidate sizes. For reference, with $k = 10$ and 100 candidates, it would still be 7 elimination rounds, but with a total of 12 disambiguations of at most 10 candidates each.

[14]Please note that we perform our evaluation on 581 documents from the AIDA dataset due to a flat multiplier of 98 costing unnecessary amounts of time and electricity on our limited hardware.

A.3 Candidate Representation (Pairwise Effects)

We investigated effects of single candidate representation types within our paper. We considered it interesting to have a look at pairwise combinations thereof as well to verify to what extent adding more information could yield better results – as would be an initial human intuition.

In Table 4, we evaluated AGNUS on pairwise combinations of candidate repsentation types to verify effects as well as the extent of increased information content on results. We note that disambiguating based on meaningful IRIs, such as from Wikipedia (e.g. https://en.wikipedia.org/wiki/Mika_(singer)), yields the best scores regardless of representation it may be combined with. Any further representation type worsens results, seemingly indicating that highly-defining compact representations may yield best results.

Types by themselves return mixed results, slightly improving upon description-based candidates, but deteriorate label-based results slightly. This may be due to the high overlap among candidates for this representation, potentially causing confusion upon disambiguation and yielding worst results (7.0%) in our experiments. Adding labels (7.3%) or descriptions (7.7%) to types increases candidate information, decreasing ambiguity and leading to improved results. Labels as an entity characteristic by themselves (7.43%) are relatively ambiguous, but benefit slightly from further information in the form of descriptions (7.44%). Overall, top scores are reached with IRI representations regardless of other combined information – actually suffering from any additional representations

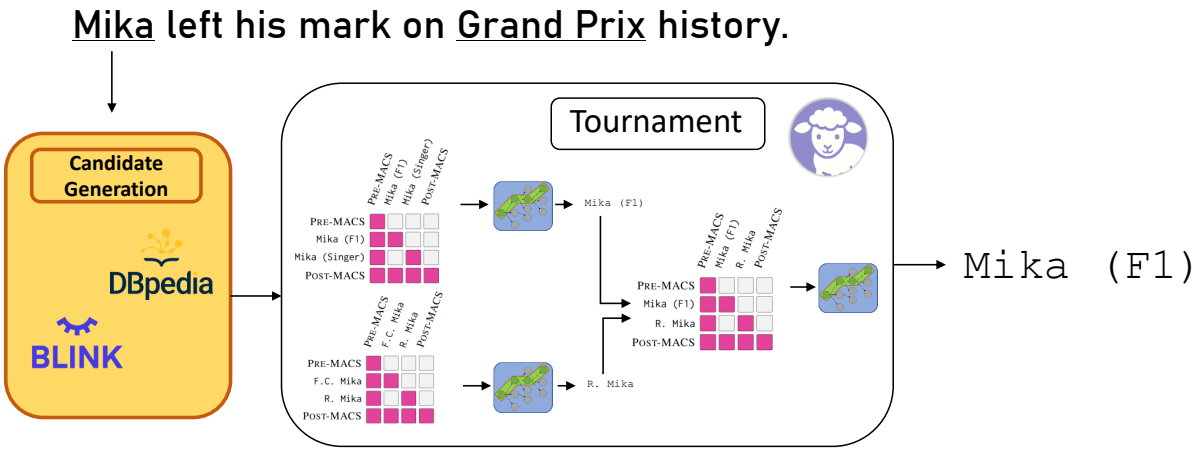

Figure 5: Agnus Hierarchical Elimination Tree: When context window exceeds LLM capabilities, Agnus makes use of a hierarchical elimination tree, splitting the disambiguation task into smaller ones, each of size k where k is the number of allowed candidates to not exceed the context window.

(by itself: 8.67%, with type(s): 8.55%, with description: 8.54%) –, most notably suffering from labels (7.63%).

A.4 Notes on Baseline Experiments

Do note that in the case of *"w.o. ACDC"* (without constrained decoding), we apply fuzzy matching between candidate representations for both predicted and expected values, ranking similarity for the sake of comparison fairness and picking the highest-overlap-similarity candidate as a match. Just using the results as-is for a "baseline" comparison seemed disingenuine as "exact matching" criteria would put baseline results very close (if not exactly) to 0 in most cases.

Applying hard-prompting based finetuning to our employed suite of large language models would likely alleviate the effects to a certain degree, but would simultaneously render the comparison invalid due to comparing our zero-shot model to a 1-shot baseline, therewith having only limited expressivity over our existing ED evaluation table (Table 1).

Due to similar reasons, our baseline without candidates still uses matching to candidates (it did not see or produce) rather than dryly applying an exact matching scheme, therewith heightening the likelihood of correct results. Hence, we urge readers to not overestimate baseline performance.