Research Square

Preprints are preliminary reports that have not undergone peer review.
They should not be considered conclusive, used to inform clinical practice,
or referenced by the media as validated information.

How Far Can Exact-Match Candidate Generation Take Arabic Entity Linking? Layered Surface-Form Coverage in Wikidata and Arabic Wikipedia

Anes Abdelouahab Mahammedi

mahammedi.anesabdelouahab@cuniv-naama.dz

University of Naama, University of Naâma

Abdelghani Bouziane

University of Naama, University of Naâma

Yasser Yahiaoui

University of Naama, University of Naâma

Djelloul Bouchiha

University of Naama, University of Naâma

Research Article

Keywords: entity linking, candidate generation, Arabic natural language processing, Wikidata, knowledge base coverage, Arabic dialects

Posted Date: November 16th, 2026

DOI: https://doi.org/10.21203/rs.3.rs-11323391/v1

License: This work is licensed under a Creative Commons Attribution 4.0 International License. Read Full License

Additional Declarations: No competing interests reported.

Abstract

Arabic entity linking loses a large share of mentions before disambiguation begins. We measure that loss with a layered audit of the public Arabic surface-form resources (Wikidata labels and aliases, ParaNames, Wikipedia titles, redirects and a held-out anchor dictionary). The instrument is exact-match lookup over 200,000 Arabic Wikipedia link mentions, which attributes every retrieved entity to a resource.

Only 17.7% of the entities with an English Wikipedia article carry an Arabic label, and almost all of them have an Arabic article: Wikidata's Arabic naming stops where Arabic Wikipedia stops. For entities that have one, all resources combined retrieve the correct entity for 74.4% of the mentions that differ from their target's title, on a benchmark that draws each mention–entity pair once. Weighted by frequency in text, the figure is 86.8%. At rank 10 the shortfall is coverage, not ranking. Anchor text supplies almost all of it, and an identity predicts the ceiling from dictionary size exactly: the share of pairs the dictionary has recorded, plus a small constant on the rest. Stripping the definite article is the most valuable normalisation step. A hand analysis of the failures separates unrecorded name forms, which dominate for persons, from benchmark artefacts. The cheap remedies it suggests recover 0.1 to 0.4 points; character n-gram and dense string matchers recover 66% and 72% of the shortfall at 100 candidates per mention. On Mewsli-9 Arabic the same resources match the published alias-table baseline. Resources and code are released.

1. Introduction

An entity linking system that never proposes the correct entity cannot recover it later, however strong its disambiguator. Candidate generation therefore sets a hard upper bound on end-to-end accuracy. For Arabic the bound has been measured in aggregate: Botha et al. (2020) report an alias-table recall@10 of 0.93 on Arabic WikiNews. It has not been decomposed. Which resource supplies which share of it, how much survives on mentions that are not already canonical names, and whether the shortfall is retrieval or ranking are, to our knowledge, unmeasured.

The Arabic literature has concentrated on the stages either side of it. Recognition has a decade of corpora and a shared task series (Hamad et al., 2025; Jarrar et al., 2023; Jarrar et al., 2022), and disambiguation attracts model papers. Between them, the step that turns a mention string into a set of candidate entities is usually a dictionary lookup mentioned in one line of a system description. In Arabic that step is unsafe for three reasons that compound. The first is knowledge base coverage: Wikidata (Vrandečić & Krötzsch, 2014) is language-agnostic by design, but its labels are not, and an entity exists whether or not anyone has written its name in Arabic. The second is orthographic variation. Arabic writes hamza, alef maksura and teh marbuta inconsistently, omits diacritics, and attaches prepositions, conjunctions and the definite article to the following word, so one entity surfaces as □□□□ ,□□□□ and □□□ in ordinary prose. The third is alias sparsity: a dictionary-based generator needs alternative names, and the supply of Arabic alternatives differs between resources in ways that have not been quantified.

We treat candidate generation as a layered resource question. Starting from the Arabic label Wikidata stores for an entity, we add one resource at a time: Wikidata aliases, ParaNames, Arabic Wikipedia titles, redirects, and a dictionary of anchor text held out from a disjoint half of the encyclopaedia. At each layer we report how many mentions produce no candidate and how often the correct entity appears at any rank, at rank 1 and at

rank 10. Exact lookup is the instrument because it attributes every retrieved entity to the resource that supplied its name, which a similarity matcher cannot do. It is also what deployed systems run: the alias table behind mGENRE (De Cao et al., 2022) and the one behind the strongest published Arabic candidate figures (Botha et al., 2020) are built from the sources layered here. Section 4.8 measures what string matchers add on top of it.

Contributions.

1. A non-circular measurement of Arabic coverage in Wikidata over English Wikipedia entities, beside 7 reference languages, showing that Arabic labels exist almost only where Arabic articles do.
2. A 200,000-mention benchmark with a held-out anchor split; an order-independent decomposition of 5 resources; the ceiling as a function of dictionary size with an exact seen-pair identity; and its uncertainty from rotating the split, under pair and occurrence weighting.
3. A three-way decomposition of candidate failure, overall and by entity class; a hand classification of 299 failures with every remedy it suggests tested; and the share of the shortfall that lexical and dense string matching recover.
4. Evidence, in Online Resource 1 (Section S1), that Wikidata's Arabic dialect label counts are largely an import artefact.

Section 2 reviews related work. Section 3 describes the resources, the benchmark and the metrics. Section 4 reports results, and Sections 5 to 7 discuss implications, limitations and conclusions.

2. Related work

2.1 Entity linking and the candidate stage

Entity linking was framed from the outset as two stages: search for candidates, then disambiguate among them. The works that established the task over Wikipedia (Bunescu & Paşca, 2006; Cucerzan, 2007) built their candidate sets from anchor text and redirects. The link-probability and relatedness features introduced shortly after (Milne & Witten, 2008) remain the backbone of dictionary-based generators, and the cross-lingual dictionary of Spitkovsky and Chang (2012) extended the construction to strings in any language. AIDA (Hoffart et al., 2011) added collective disambiguation and supplied the benchmark on which most later systems report; the standard survey (Shen et al., 2015) organises the field along the same two stages. Hachey et al. (2013) showed that comparison between systems had concentrated on disambiguation accuracy, so that the share of performance owed to candidate search was invisible; once search was controlled for, the published differences narrowed considerably. We ask their question of a language rather than of a system.

Dense retrieval offers a way around the dictionary. BLINK (Wu et al., 2020) embeds mention context and entity description in a shared space and retrieves without any alias table, and retriever–reader architectures such as ReLiK (Orlando et al., 2024) make that practical on one GPU. A multilingual bi-encoder can match an Arabic mention against a representation built from other languages, which is how the dual encoder of Botha et al. (2020) exceeds its own alias table on Arabic. The ceiling we report bounds lookup over Arabic surface forms.

2.2 Arabic entity recognition resources

Arabic named entity recognition (NER) is well supplied, and Shaalan's (2014) survey documents the rule-based and feature-engineered era that preceded the current corpora. ANERcorp (Benajiba et al., 2007) established an early newswire benchmark of roughly 150K tokens, and AQMAR (Mohit et al., 2012) added hand-annotated Wikipedia text. Wojood (Jarrar et al., 2022) contributed roughly 550K tokens of Modern Standard Arabic (MSA) and Levantine dialect annotated with 21 nested entity types, later extended by a shared task series (Jarrar et al., 2023). More recent work widens the coverage. Konooz (Hamad et al., 2025) spans 16 dialects across 10 domains with about 777K tokens annotated under the Wojood guidelines, and dialect-specific corpora exist for Algerian (Dahou & Cheragui, 2023) and Moroccan Arabic (Moussa & Mourhir, 2023). All of these stop at the span. None assigns a knowledge base identifier, so none can be used to evaluate linking without further annotation. One recent release does link dialectal mentions: ELNER-DZ (Bouguettoucha & Djouablia, 2025), an Algerian Arabic dataset linked to Wikidata and released with a master's thesis; we leave evaluation on it to future work.

2.3 Arabic entity linking and disambiguation

Arabic entity linking itself is not new. AIDArabic (Yosef et al., 2014) extended the AIDA framework to Arabic and identified the obstacle we quantify here: the sparsity of Arabic Wikipedia leaves the entity catalogue too thin. Its authors mitigated it by importing entities through interwiki links, then with EDRAK, an automatically generated Arabic entity resource (Gad-Elrab et al., 2015a). A later method fuses multilingual resources, machine translation and transliteration to synthesise disambiguation systems for resource-poor languages (Gad-Elrab et al., 2015b). Those systems treated the sparsity as an obstacle to route around; none measured what it costs at the candidate stage. Kaffee et al. (2017) measured the distribution of Wikidata labels across languages from a 2017 dump and found it far from proportional to speakers. Section 4.1 narrows that measurement to the entities relevant for linking and adds aliases.

Resources that do carry identifiers are recent and few. ArabNED (Nounou & Sabty, 2026) provides roughly 10,000 Arabic named entities with disambiguation candidates over ANERcorp and AQMAR. ANERD (Alotaibi & Menai, 2026a, 2026b) is larger, with 502,756 sentences across 3 difficulty tiers, and links entities to Arabic Wikipedia with explicit NIL marking; its data descriptor reports 2 annotators with adjudication and a title agreement of 0.81. Its disambiguation baseline is the closest published observation to ours. A frequency-ranked mention–entity dictionary from the training split retrieves the gold entity within 30 candidates for 85.4%, 68.5% and 78.2% of test mentions on the 3 tiers, and 100 candidates add at most 0.09 points. Once retrieved, the reranker places the entity first for 99.9%, 99.7% and 89.4% (Alotaibi & Menai, 2026a). Coverage rather than reranking is therefore the bottleneck on the middle tier, which is the shape of Section 4.2 observed inside one corpus with one resource. Online Resource 1 (Section S2) reports our audit of the corpus. Both resources are Modern Standard Arabic newswire or encyclopaedic text, and neither was designed to isolate candidate generation from disambiguation.

2.4 Multilingual entity linking

The multilingual literature has repeatedly found that low-resource languages fail at retrieval rather than at ranking. Botha et al. (2020) train a dual encoder over 100 + languages and 20M entities and release Mewsli-9, weighted towards entities without English Wikipedia pages. For its Arabic portion (1,468 documents, 7,367 mentions) they report an alias-table baseline of recall@1 0.89 and recall@10 0.93, and a dual-encoder result of 0.92 and 0.98. These are the only Arabic candidate-recall figures on a public benchmark we are aware of;

cross-lingual wikification systems that include Arabic (Pan et al., 2017; Tsai & Roth, 2016) report linking accuracy instead. The figures fix the level but do not say which resources produce it, and the alias table pools names from 104 languages. Section 4.6 re-runs our layers on the same data. Mewsli-X (Ruder et al., 2021) keeps Arabic among 11 mention languages as a dense-retrieval task with English-only fine-tuning. DaMuEL (Kubeša & Straka, 2023) provides Wikipedia text with Wikidata links for 53 languages as a training resource. Neither reports what any Arabic surface-form resource contributes. Autoregressive formulations such as mGENRE (De Cao et al., 2022) need no explicit candidate set. Their alias table is nevertheless built from exactly the sources we layer, and enabling it is what puts mGENRE ahead of the prior state of the art on 2 of its 3 benchmarks.

2.5 Alias resources and candidate generation

Outside Arabic, the standard candidate generator is a mention–entity dictionary built from Wikipedia anchor text, optionally augmented with redirects and knowledge base aliases. ParaNames (Sälevä & Lignos, 2024) supplies 140 million names for 16.8 million entities across 400 + languages, derived from Wikidata (release v2024.05.07.0, 829,562 Arabic names in our copy). Its authors note that it uses main labels only and omits aliases, and its published gazetteer evaluation covers 10 languages that do not include Arabic. The cross-lingual literature has reached a conclusion that parallels ours. Zhou et al. (2020) show that Wikipedia-derived candidate generation, effective for high-resource languages, degrades sharply for languages with few Wikipedia pages. Fu et al. (2020) find in error analysis of low-resource cross-lingual linking that candidate generation rather than ranking is the key obstacle, and that generators tuned on Wikipedia fail to generalise to news and social media text. Arabic is not low-resource by page count, yet Section 4.2 finds the same shape. No study reports what any of these resources contributes to Arabic candidate recall, separately or in combination.

2.6 Arabic preprocessing

Arabic normalisation is well studied as a preprocessing question. CAMeL Tools (Obeid et al., 2020) implements the operations that have become standard: Unicode normalisation, dediacritisation, tatweel removal, alef and alef maksura unification, and teh marbuta unification. Studies exist of their effect on text classification (Ayedh et al., 2016), machine translation (El Kholy & Habash, 2012) and parsing (Marton et al., 2013). We are not aware of a study that measures their effect on entity candidate retrieval, where the quantity at stake is whether two strings collide in a dictionary rather than a classifier’s decision boundary.

3. Method

3.1 Surface-form sources

We assembled every Arabic string that maps to a Wikidata entity from 5 sources (Table 1): Wikidata labels, Wikidata aliases, ParaNames, Arabic Wikipedia titles and redirects, and Arabic Wikipedia anchor text. Wikidata Arabic labels and aliases were streamed from the QLever SPARQL engine (Bast & Buchhold, 2017) over the full Wikidata graph, restricted to items and excluding lexemes, forms, senses and properties. Arabic Wikipedia titles and redirects were parsed from the page, page_props and redirect SQL dumps. Disambiguation pages were excluded using the disambiguation page property, and a parenthetical disambiguator produces an

additional stripped form rather than replacing the full title. Anchor text was extracted from the pages-articles XML dump by scanning wikitext for internal links, discarding links whose target contains a namespace colon.

Table 1
Surface-form sources.

Source	Surface forms	Entities	Distinct strings
Wikidata label (ar)	4,117,659	4,117,631	3,850,532
Wikipedia anchor text	2,031,933	824,857	1,761,865
Wikidata alias (ar)	1,517,917	1,226,813	1,479,193
Article title	1,294,556	1,294,556	1,294,556
Redirect	1,096,526	661,302	1,096,526
ParaNames (ar)	829,562	829,562	—
Title, disambiguator stripped	267,457	267,457	182,263
Redirect, disambiguator stripped	146,471	133,224	108,725

Two further sources were tested and left out. Egyptian Arabic (arz) labels, 1.9 million of them (Online Resource 1, Section S1), add 0.08 points of recall@∞ on top of the full stack (Section 4.7), because almost every entity they name already carries a Modern Standard Arabic label. Arabic DBpedia and NER gazetteers were not tested; both derive from the same Wikipedia and Wikidata sources already layered here.

Three figures in Table 1 already carry information. Wikidata stores one label per language, so the label row is one string per entity. Aliases exist for 1,226,911 entities, but 460,644 of those carry no Arabic label and are, on inspection, Latin-script surnames filed under the Arabic code (Section 4.1); among labelled entities the alias rate is 18.6%. ParaNames carries exactly 1.00 names per entity, confirming its authors' statement that it is built from main labels without aliases. Its 829,562 entities are the Wikidata items typed as person, location or organisation, after removing 53,264 rows duplicated across those types. Entity counts here are slightly below those of Section 4.1 because the pool drops surface forms longer than 120 characters (791 labels, 98 aliases).

Table 2 defines the cumulative resource layers built from these sources.

Table 2
The six cumulative resource layers and one leave-one-out configuration. Orthographic normalisation is an independent dimension (Section 3.3) rather than a layer; every result table names the profile it uses.

Layer	Sources added
S0	Wikidata Arabic label
S1	+ Wikidata Arabic aliases
S2	+ ParaNames Arabic
S3	+ article titles (and stripped)
S4	+ redirects (and stripped)
S5	+ held-out anchor dictionary
S5-PN	S5 with ParaNames removed

3.2 Evaluation set

Our first plan was to measure candidate recall on ANERD (Alotaibi & Menai, 2026a, 2026b), the only Arabic corpus combining entity spans, gold links and NIL marking at scale. An audit, reported in Online Resource 1 (Section S2), found three problems. 7.38% of its linkable gold annotations name Arabic Wikipedia titles that do not exist as articles or redirects; 626 mentions carry the month August as a PERS entity; and its locations are dominated by Italian geography. The data descriptor describes a validation pass over the links but no check of the targets against the encyclopaedia. Measuring candidate recall against those links would measure the corpus rather than the resources, so we built the benchmark below and, for external validity, evaluate on Mewsli-9 in Section 4.6.

Every internal link in Wikipedia is a mention paired with an entity, which is the standard way multilingual linking data is built (Botha et al., 2020; Kubeša & Straka, 2023; Pan et al., 2017). Extraction over the Arabic dump yielded 30,093,914 link instances and 5,805,493 distinct (anchor, target) pairs. Targets resolved to Wikidata identifiers for 72.7% of distinct pairs and 92.1% of instances; the rest are dominated by disambiguation pages and redlinks, both correctly excluded. The bucket counts below aggregate by resolved entity, so a redirect and its destination collapse into one row. This is why the two buckets sum to 4,146,878 rather than the 4,220,600 resolved pairs.

Building both the anchor dictionary and the evaluation mentions from the same links would be circular, since an anchor pointing at an article is itself evidence that the anchor names that article. We therefore split source articles by MD5 hash into two halves. Bucket 0 (2,031,933 pairs, 824,857 entities) supplies the anchor dictionary used as layer S5, and bucket 1 (2,114,945 pairs, 827,279 entities) the evaluation mentions. The split is by source article rather than by pair, so a frequent (anchor, entity) pair can occur in both halves. This is intended: it is exactly what measures the dictionary’s value on mentions it was not derived from, and Sections 4.2 and 4.3 report the consequence.

From the 2,114,945 distinct (anchor, entity) pairs of bucket 1 we drew 200,000 without replacement, with probability proportional to each pair's link count, covering 143,533 distinct entities. Because a pair can be drawn at most once, the sample does not reproduce the frequency distribution of running text. Bucket 1 holds 13,897,591 link occurrences, so occurrence-proportional sampling would draw a pair once per 69.5 links. The 19,274 pairs linked more often than that are 0.9% of the distinct pairs but carry 53.0% of the occurrences; 18,770 of them are in the sample, each once, where they form 9.4% of it. Frequent pairs are capped, and the sample weights the tail of the pair distribution more heavily than text does. We call it the pair sample; Section 4.2 reports an occurrence sample beside it, in which a pair appears in proportion to its frequency. In 52.2% of the sample the mention string equals the link target as written in the wikitext, between the 39.6% of distinct pairs and the 59.1% of occurrences that do. We call these **literal** mentions; the target may itself be a redirect, which is why 2.9% of them still retrieve no candidate until redirects are indexed. Literal mentions are largely recoverable by exact lookup, so headline results are reported on the remaining 95,699 **non-literal** mentions, with the full set and the literal subset alongside.

One property of the benchmark bounds what it can measure. Every gold entity is the target of an Arabic Wikipedia link and therefore has an Arabic article and, with few exceptions, an Arabic label. All recall figures in Sections 4.2 to 4.8 are consequently conditional on the entity having an Arabic article: they measure whether the resources hold the *name form* used in text for an entity Arabic Wikipedia already describes. The larger gap of Section 4.1, the entities that have no Arabic name at all, lies outside the benchmark and is not reduced by anything measured on it.

3.3 Normalisation profiles

Six cumulative profiles are used (Online Resource 1, Table S5). *raw* applies Unicode NFKC and whitespace collapse; *dediac* removes diacritics and tatweel; *alef* unifies alef variants and alef maksura; *teh* unifies teh marbuta with heh; *punct* strips punctuation. *noal* strips a word-initial definite article alone or with one attached proclitic (□□□ □□□□ □□□□ □□□□ □□□). The two used throughout are punct (the first five) and noal (all six); the ablation in Section 4.4 uses every intermediate profile. The first five follow the conventions of CAMeL Tools (Obeid et al., 2020). The sixth is not part of the stack shipped by Arabic NLP toolkits, but it has been standard in Arabic information retrieval since the light stemmers of Larkey et al. (2002, 2007), whose Light10 strips this prefix set together with □□ and a bare □. Our rule removes a leading □□ optionally preceded by one of □ □ □ □, only when at least two characters follow; it leaves a bare proclitic (□□□□ □□□□□) and the contraction (□□□□□□□□) □□ in place. Section 4.4 measures what the fuller set adds. The rule over-segments: after alef unification, □□□□□□□ becomes □□□□□□□ and is cut to □□□□□. Since it is applied to mentions and pool alike, such cases still match themselves, at the cost of occasional collisions with unrelated keys.

3.4 Ranking prior

Candidates sharing a normalised key are ordered by the standard mention-conditioned prior P(e|m) (Milne & Witten, 2008). This is the number of links in bucket 0 whose anchor text normalises, under the profile in force, to the key and whose target is the entity. Candidates with no such count (keys that never occur as anchor text, or entities never linked from that key) are ordered after them by a popularity back-off P(e). This is the entity's total link count in bucket 0, itself backed off to half the Wikidata sitelink count. Both quantities come from the resource half, disjoint from the evaluation half. No trained model is involved; the object of study is the resource.

We also report results under the popularity prior alone, because the difference is instructive. It leaves recall@∞ unchanged and lowers recall@1 at the full stack by 5.9 points on non-literal mentions (51.11 against 57.00). It also creates one apparent finding, that the anchor dictionary lowers rank-1 accuracy on easy mentions by 4.6 points, which shrinks to 0.7 under P(e|m) and is a property of the weaker ranker (Section 4.2). The back-off weight is not critical: 0, 0.5 or 1 changes recall@1 by 0.45 points in total.

3.5 Metrics

For every mention we record whether the candidate set is empty, whether the gold entity is present at any rank, and if so at which rank. Four quantities follow and partition the mentions exhaustively. The **empty rate** is the share with no candidate, a resource gap. **Absent with candidates** is the share with candidates that exclude the gold entity, a resource gap a system cannot detect at run time. **Recall@∞** is the share whose gold entity is present at some rank; it is the ceiling, which no ranker can exceed. **Recall@k** is the share whose gold entity is within the top *k* under our prior. At k = 1 it is a baseline that any trained disambiguator should beat, reported because it is what a lookup system delivers with no further modelling. The **ranking gap** is recall@∞ − recall@10: mentions whose gold entity was retrieved but ranked below 10. It isolates ranking failure from coverage failure, which the naive quantity (100 − empty) − recall@10 conflates.

Both priors are derived from bucket-0 anchor counts, so recall@k at layers that contain no anchor text is not a Wikidata-only quantity; recall@∞ and the empty rate are unaffected, and they carry the ceiling claims. Confidence intervals are 95% bootstrap intervals with 1,000 resamples of entities rather than mentions, since mentions of the same entity are not independent. With 63,923 entities behind the 95,699 non-literal mentions the clustering is weak: 74.15–74.73 at the full stack against 74.17–74.73 at mention level. Section 4.2 adds, for the occurrence sample, intervals that resample source articles, the unit of the split, and the spread of the ceiling across 16 held-out buckets.

4. Results

4.1 How much of Wikidata exists in Arabic

We first bound the problem independently of any Arabic resource. Taking the set of entities that have an English Wikipedia article as the denominator, a choice made outside Arabic and therefore not circular, we ask how many carry any Arabic name (Table 3).

Table 3

Coverage of entities that English Wikipedia covers, for Arabic and seven reference languages, in order of label share. Denominator 7,036,941: main namespace only, excluding items typed as disambiguation page, list, category, template or project page. “Label” means a label under the language’s primary code (ar is Modern Standard Arabic; dialect codes are treated separately in Online Resource 1, Section S1). Queried 21–25 September 2026.

Language	Label in Wikidata	Share	Wikipedia article	Share	Labels per article
Persian	1,007,359	14.3%	957,928	13.6%	1.1
Turkish	1,139,824	16.2%	511,560	7.3%	2.2
Arabic	**1,244,714**	**17.7%**	**1,044,081**	**14.8%**	**1.2**
Japanese	1,390,432	19.8%	701,262	10.0%	2.0
Russian	2,012,660	28.6%	1,161,115	16.5%	1.7
German	3,649,379	51.9%	1,601,273	22.8%	2.3
Spanish	3,912,895	55.6%	1,412,538	20.1%	2.8
French	4,210,100	59.8%	1,862,701	26.5%	2.3

82.3% of the entities an English-language system can address have no Arabic name in Wikidata at all. Arabic is not an outlier: Persian and Turkish sit at or below it and Japanese just above. Script does not explain the ordering, since Latin-script Turkish is second lowest, Cyrillic Russian is mid-table, and the three languages above 50% are those with the largest Wikipedias and the highest label-to-article ratios. The last column is the informative one. Arabic has more articles than Japanese or Turkish yet fewer labels than Japanese, because it carries only 1.2 labels per article: an Arabic label exists almost only where an Arabic article exists. French, Spanish and German carry 2.3 to 2.8 labels per article, names for entities with no article in that language, the product of label-import projects. The Arabic shortage is therefore a shortage of labelling beyond the articles rather than of articles, the kind of gap that import from redirects, other-language labels or gazetteers can close (Section 5). It bounds any method that searches the knowledge base with an Arabic string. It does not bound cross-lingual methods that match an Arabic mention against representations built from other languages, which is how the dual encoder of Botha et al. (2020) exceeds its own alias table. Nor is it measured by the benchmark of Section 3.2, whose entities all have Arabic articles. The ceilings of Sections 4.2 to 4.8 describe the entities inside the Arabic-named share; for a mention of an entity outside it the candidate set is empty by construction, so the two gaps compound.

The restriction to articles matters. Counting sitelinks of every namespace raises the denominator to 10,345,174 and inflates the Arabic shares to 24.6% and 19.6%, and keeping disambiguation and list pages gives 7,537,360 and 16.8%. All figures here exclude both; the Arabic count of 1,044,081 is consistent with the 1,294,556 articles in our snapshot.

Within the Arabic portion, alternative names are scarcer still. Of the 1,287,704 entities that have both a main-namespace Arabic article and an Arabic label, only 182,868 (14.2%) carry any Arabic alias in Wikidata; over all 4,118,422 Arabic-labelled items the rate is 18.6%. 85.8% of the entities that Arabic Wikipedia describes have exactly one Arabic name on record. A further 460,644 items carry an Arabic-tagged alias with no Arabic label

at all. A random sample of 300 of them, checked against the live Wikidata API on 25 September 2026, shows what they are. They are family-name entities with no Arabic article and usually no sitelink at all, whose Arabic-tagged alias is the Latin-script surname itself (Q37558626 *Castilho*, Q37465735 *Schranz*). One string was filed under many language codes by an import; a direct count on the QLever index gives 460,575 such items. Because these strings are not Arabic they can match nothing in Arabic text, and the 18.6% over labelled entities is the rate that bears on candidate generation.

Dialect codes

Wikidata also carries labels under Arabic dialect codes, and the raw counts look substantial: 351,466 items with an Algerian Arabic (arq) label, 154,033 Tunisian (aeb), 520,365 Moroccan (ary), 1,915,675 Egyptian (arz). Online Resource 1 (Section S1) filters them for script, difference from the Modern Standard Arabic label, date entities and Wikimedia-internal pages. The Algerian count falls to 538: 152,881 labels are not in Arabic script, 84,369 repeat the MSA label, and 113,669 of the 114,216 that differ from it are calendar dates from one automated import. Tunisian falls to 80 and Moroccan to 12,720, while Egyptian, with a large bot-built Wikipedia of its own, keeps 925,503. Per-language label counts in Wikidata are not evidence of language coverage until they are filtered. Section 4.7 tests the Egyptian labels as a further layer.

4.2 Candidate recall: ceiling, layers and order

Table 4
Non-literal mentions (n = 95,699), noal normalisation, P(e|m) prior. The first three numeric columns partition the mentions: empty + absent-with-candidates + recall@∞ = 100. Figure 1 plots the decomposition.

Layer	Resource added	Empty %	Absent w/ cands %	R@∞	95% CI	R@1	R@10	Gap@1	Gap@10
S0	Wikidata Arabic label	56.81	21.00	22.20	21.9–22.5	18.75	22.12	3.45	0.08
S1	+ Wikidata Arabic aliases	46.49	19.75	33.76	33.4–34.1	29.15	33.67	4.61	0.09
S2	+ ParaNames	46.20	19.56	34.24	33.9–34.5	29.45	34.11	4.79	0.12
S3	+ article titles	45.09	18.81	36.10	35.8–36.4	30.63	35.92	5.46	0.18
S4	+ redirects	36.18	16.92	46.90	46.6–47.2	40.51	46.71	6.39	0.19
S5	+ anchor dictionary	18.23	7.33	**74.44**	**74.2–74.7**	57.00	71.89	17.44	2.55

Under punct normalisation the ladder of Table 4 ends at 71.99 with the same ordering of layers; under the popularity-only prior recall@1 at S5 is 51.11 and the rank-10 gap 3.95 (Online Resource 1, Section S4).

Candidate sets stay small, so the ceiling is not bought with long lists (Online Resource 1, Table S6). Over non-literal mentions with at least one candidate, the median set has 1 or 2 entries at every layer and the 90th

percentile 11–12 through S4 and 29 at S5. The mean rises from 2.05 at S0 to 11.21 at S5 because a handful of generic keys carry thousands of candidates (the largest, 5,688).

Over all 200,000 mentions, S5 reaches recall@∞ of 87.77 under noal (95% CI 87.63–87.92); the literal subset is fully covered at S5, so 87.77 is 100 × (104,301 + 71,238) / 200,000. Section S4 of Online Resource 1 gives the full grids.

The ceiling and its split between coverage and ranking

With every resource combined and a dictionary from half the encyclopaedia, the ceiling on non-literal mentions is 74.4%: 18.2% retrieve nothing, and 7.3% retrieve candidates that do not include the correct entity (Fig. 1). Neither can be repaired downstream. How the remainder splits between coverage and ranking depends on the ranking line. At rank 10, the depth a disambiguator typically re-ranks, the correct entity is retrieved but buried for 2.55% of mentions, 10% of the coverage failure. At rank 1 the gap is 17.4 points, 68% of it. A lookup system with no disambiguator therefore loses about as much to ranking as to coverage; a system that re-ranks 10 candidates loses 10 times more to coverage.

Order of addition

Table 4 reads as if redirects contribute 10.8 points of recall@∞ and anchor text 27.5, but that is an artefact of adding redirects first. Table 5 removes the order.

Table 5
Each resource alone on top of S0, and each removed from the full stack S5. Non-literal, noal, recall@∞.

Resource	Alone over S0	Δ vs S0	Removed from S5	Δ vs S5
(S0 / S5 reference)	22.20	—	74.44	—
Anchor dictionary	72.29	+ 50.09	46.90	−27.54
Redirects	43.93	+ 21.73	73.37	−1.07
Wikidata aliases	33.76	+ 11.56	74.27	−0.17
Article titles	25.10	+ 2.90	74.20	−0.24
ParaNames	22.89	+ 0.69	74.44	0.00

Anchor text alone reaches 72.29; every other resource combined adds 2.15 points on top of it. Redirects, worth 21.73 points on their own, cost 1.07 when removed from a stack that already contains anchors; aliases, worth 11.56 alone, cost 0.17. The three usage-derived resources substitute for one another almost completely. The two canonical-name resources, titles and ParaNames, add little in either configuration because 77.6% of titles and 95.4% of ParaNames pairs are byte-identical to the Wikidata label already present at S0. The practical reading is a conditional. With Arabic Wikipedia anchor text, nothing else matters much; without it, redirects are the best substitute, then aliases, and canonical-name resources are not worth the effort.

Seen and unseen pairs

The split is by source article, so a (mention, entity) pair that occurs in several articles can appear in both halves, which is how a mention–entity dictionary works. The out-of-domain result of Section 4.6 shows the dictionary generalising, but the headline is an average of two regimes. Under the normalised keys the dictionary uses, 71.1% of non-literal evaluation pairs are recorded in it and every one is retrieved. On the other 28.9%, recall@∞ at S5 is 11.6, and the identity 71.1 + 28.9 × 0.116 = 74.4 holds exactly (on verbatim strings, 65.9% and 25.1). The seen share is a property of the split, so the ceiling is a function of dictionary size rather than a constant of the language.

The ceiling as a function of dictionary size

To measure that function we re-split the encyclopaedia into 16 article buckets by hash and drew a second evaluation sample of 100,000 mentions (46,533 non-literal) from one bucket that never enters the dictionary. The anchor dictionary was then built from 1, 2, 4, 8, 12 and 15 of the other 15 buckets. Table 6 and Fig. 2 give the result; the paper's own sample, evaluated against dictionaries from 1 to 8 even buckets, reproduces 74.44 at 8/16.

Table 6

Recall@∞ at S5 as a function of the share of Arabic Wikipedia articles supplying the anchor dictionary. *Bucket 15* is the held-out evaluation sample (46,533 non-literal mentions); *paper sample* is the 95,699 non-literal mentions of Table 4, whose dictionary can only use the eight even buckets. *Seen* is the share of pairs recorded in the dictionary under normalised keys; recall@∞ on seen pairs is 100 in every row. The last two rows are occurrence samples of 50,000 link occurrences per held-out bucket, averaged over the eight odd buckets (dictionary from the eight even buckets, the paper's setting) and over all 16 (dictionary from the other 15); ± is the standard deviation across buckets, and recall@1 was not computed for them.

Sample	Dictionary share	Dictionary pairs	Seen %	R@∞ unseen	Empty %	R@1	R@∞
bucket 15	1/16	481,454	51.7	27.98	24.13	51.70	65.20
bucket 15	2/16	794,951	60.3	21.96	21.67	54.50	69.05
bucket 15	4/16	1,287,226	68.1	16.31	18.92	57.32	73.27
bucket 15	8/16	2,038,216	74.6	11.23	16.12	60.11	77.41
bucket 15	12/16	2,633,110	77.8	8.71	14.53	61.59	79.77
bucket 15	15/16	3,025,172	79.5	7.76	13.65	62.50	81.11
paper sample	1/16	481,454	45.9	28.03	27.11	47.88	61.03
paper sample	2/16	794,262	55.2	22.37	24.40	50.85	65.19
paper sample	4/16	1,287,025	63.7	16.85	21.41	53.85	69.80
paper sample	8/16	2,031,933	71.1	11.59	18.23	57.00	74.44
occurrence, 8 odd	8/16	2,031,933	85.1	11.65	9.36	—	86.8 ± 0.3
occurrence, all 16	15/16	≈ 3.0 M	88.0	8.15	7.84	—	89.0 ± 0.3

The ceiling rises from 65.2 with anchors from 1/16 of the encyclopaedia to 81.1 with 15/16, with shrinking increments (4.2 points from 2/16 to 4/16, 1.3 from 12/16 to 15/16), so the complete dictionary would sit near 81.5. Both components move. The seen share climbs from 51.7% to 79.5% and saturates, while the ceiling on unseen pairs falls from 28.0 to 7.8, because the pairs a larger dictionary still lacks are the ones no other resource holds either. On this sample the paper's own setting gives 77.4, so the complete anchor text adds about 3.7 points and a complete dictionary about 4, and still leaves 1 non-literal mention in 5 without its entity. The bucket-15 sample sits 3 to 4 points above the paper sample at equal dictionary share (77.4 against 74.4 at 8/16). The choice of bucket is not the reason, since rotating it moves the ceiling by 0.3 points (below). The degree of capping is: the bucket-15 sample draws 100,000 of one bucket's pairs and the paper sample 200,000 of 2.1 million, so the former admits more of the head, where pairs are recorded elsewhere.

Occurrence weighting and rotation of the held-out bucket

To measure the ceiling on the frequency distribution of running text, and to separate split variance from sample variance, we re-extracted the links with their source article and rotated the held-out bucket over all 16. In each rotation we drew 50,000 link occurrences from the held-out bucket against a dictionary from the other 15 buckets. Each draw takes one (article, anchor, target) row, so a pair linked from 1,000 articles can be drawn up to 1,000 times. The paper's setting, the 8 even buckets against each odd one, was repeated the same way (Table 6, last rows; every rotation in Online Resource 1, Table S4). Under occurrence weighting 59% of mentions are literal against 52% in the pair sample. The non-literal ceiling at the paper's setting is 86.8 rather than 74.4, because 85.1% of the pairs are recorded in the dictionary against 71.1%; the identity holds here too (85.1 + 14.9 × 0.117 = 86.8). The complete dictionary adds 2.1 points; the empty rate is 9.4 and 7.8 and absent-with-candidates 3.8 and 3.2. Across the 16 held-out buckets the ceiling has a standard deviation of 0.30 (range 88.5–89.5). The bootstrap interval that resamples source articles has a mean half-width of 0.5 and the one that resamples entities 0.9, so clustering by article is the weaker dependence and the entity-level intervals used elsewhere are the conservative ones. The difference between the two samples of Table 6 is therefore the sampling scheme rather than split variance. Over all mentions the occurrence-weighted ceiling is about 94.6 at the paper's setting and 95.5 with the complete dictionary, which is where Mewsli-9 lands in Section 4.6.

Rank-1 effects and the prior

Under the popularity-only prior, recall@1 on the 104,301 literal mentions falls from 96.94 at S4 to 92.31 at S5, because the anchor dictionary inserts popular competitors above the correct entity. Under P(e|m) the same step moves it from 98.70 to 98.01. The loss is real for a system that ranks by popularity alone and negligible for one that conditions on the mention string.

4.3 Frequency bands

Entities were binned by their link count in the resource half: tail (≤ 2), upper tail (3–20), torso (21–200) and head (> 200). Figure 3 plots the ceiling by band for non-literal mentions under noal, from the same run as Table 4; Online Resource 1, Table S7, gives the figures. The anchor dictionary is a head resource. It raises the ceiling by 35.6 points for the most frequent entities and by 7.1 for the rarest, consistent with the unseen-pair analysis of Section 4.2. In the tail, 36.2% of mentions still produce no candidate after every resource is combined and the ceiling is 45.8; the mentions hardest to link are the ones no resource describes.

4.4 Normalisation ablation

Table 7 fixes the sources at S4 and uses non-literal mentions, each profile adding one operation to the one above it.

Table 7
Contribution of each normalisation operation, sources fixed at S4, non-literal mentions, same pool as Table 4. The *cum.* columns add one operation to the profile above; the *LOO* column removes each operation singly from the full noal profile (leave-one-out). The P(e|m) prior is recomputed under each profile, so that its keys match the profile's keys.

Profile	Operation added	Empty %	R@1	R@10	R@∞	Δ R@∞ cum.	Δ R@1 cum.	Δ R@∞ LOO
raw	NFKC + whitespace	45.25	33.67	38.47	38.62	—	—	—
dediac	diacritics, tatweel	43.57	34.91	39.80	39.96	+ 1.34	+ 1.24	−1.52
alef	alef, alef maksura	42.87	35.39	40.43	40.59	+ 0.63	+ 0.48	−0.65
teh	teh marbuta	42.78	35.44	40.51	40.67	+ 0.08	+ 0.05	−0.08
punct	punctuation	42.00	36.00	41.11	41.28	+ 0.61	+ 0.56	−0.68
noal	definite article	36.18	40.51	46.71	46.90	**+ 5.62**	**+ 4.51**	**−5.62**

The four operations that constitute the standard NLP normalisation stack contribute 2.66 points of recall@∞ in total. The one operation that is standard in Arabic IR (Larkey et al., 2002) but absent from that stack contributes 5.62. Teh marbuta unification, universally recommended in Arabic preprocessing, contributes 0.08 for this task.

The trade is favourable but not free. Stripping the definite article raises mean candidate-set size at S4 from 2.25 to 2.95, since it merges keys that were previously distinct. Repeating the noal-versus-punct comparison at every layer (Online Resource 1, Table S8), the operation always raises coverage and the ceiling: by 7.0 points at S2, 5.6 at S4 and 2.5 at S5, less as candidates become plentiful. Under the mention-conditioned prior it raises recall@1 at every layer as well, by 6.1 at S2 and still 1.1 at S5. Only under the popularity-only prior does the rank-1 effect turn slightly negative at S5 (Online Resource 1, Section S4). The merged keys admit popular competitors, which a prior conditioned on the mention string ranks below the entity actually linked from it. On literal mentions at S5 the operation costs 0.37 points of recall@1. Stripping the article is first a retrieval device; with a mention-conditioned prior it is not a precision cost either.

Because the cumulative ablation depends on the order in which operations are added, the last column of Table 7 removes each operation singly from the full noal profile. The ordering is identical: definite-article stripping is the largest single contributor to recall@∞ and teh marbuta unification the smallest by a factor of 70, under both analyses.

A fuller prefix rule

The noal rule strips only □□ with at most one attached proclitic. Two extensions were tested at S4 and S5 (Online Resource 1, Table S9): the full Light10 prefix set of Larkey et al. (2007), namely □□ ,□□□ ,□□□ ,□□□ ,□□□ ,□□ and a bare □, and that set with the bare proclitics □ □ □ □ as well. Each is applied only when at least 2 characters follow (3 for a bare proclitic). The fuller set raises the ceiling by a further 2.5 points at S4 (46.90 to 49.37) and 1.4 at S5 (74.44 to 75.84) at no cost on literal mentions. Adding bare proclitics raises it by another 1.2 and 0.9 but begins to cut real names (□□□□□ ,□□□□ ,□□□□□ lose their first letter) and costs 0.5 points of literal recall@1. Under the Light10 set the article step is worth 8.1 points of recall@∞ at S4 rather than 5.6, so the ranking of Table 7 is unchanged and strengthened. All other results use the narrower rule, so the headline ceiling is conservative by 1.4 points.

4.5 Entity classes

Entities were classified by P31 followed by the transitive closure of P279 to 5 root classes. Because *state* is a subclass of *organization* in Wikidata, countries and administrative units satisfy both the place and the organisation root (4,593 gold entities). Ties are resolved by the priority person > place > organisation > work > event. 38.6% of non-literal mentions fall outside all five roots and form the class *other*. Table 8 gives the decomposition at S5. (In the event row, absent-with-candidates and recall@1 both equal 21.92, a coincidence of counts.)

Table 8
Failure decomposition by entity class, S5, noal, P(e|m) prior, non-literal mentions. Empty + absent-with-candidates + recall@∞ = 100.

Class	n	Empty %	Absent w/ cands %	Coverage failure	R@∞	R@1	R@10	Gap@1	Gap@10
Event	4,653	13.54	21.92	35.46	64.54	21.92	42.73	42.62	**21.81**
Work	10,046	18.34	6.71	25.05	74.96	57.08	72.46	17.88	2.50
Organisation	13,431	17.30	6.04	23.34	76.66	58.65	74.78	18.01	1.88
Place	18,658	13.50	4.58	18.08	81.92	61.61	80.26	20.30	1.66
Other	36,945	19.63	7.63	27.26	72.74	58.77	71.36	13.97	1.38
Person	11,966	24.08	6.95	31.03	68.97	56.11	68.08	12.86	0.89

Persons

Persons have the highest empty rate of any class (24.1%), the highest coverage failure of any class but events (31.0%, against 35.5% for events) and the lowest rank-10 gap (0.9). When a person mention retrieves the right entity, the prior ranks it well. The problem is that 24% of person mentions retrieve nothing, because human name variation (word order, kunyas, dropped particles, transliteration choices) is recorded almost nowhere. To test how much of this exact lookup is leaving on the table, we tried two partial-match rules on the 3,713 person mentions whose correct entity is absent at S5. They match against the last token, or the first and last tokens, of every surface form in the Wikidata, title and redirect layers. The index was built over entities typed as person (an oracle, since it uses the gold class) and over all entities (what a system could do); Online

Resource 1, Table S10, gives the full result. Surname lookup recovers the correct entity for 50.9% of the person mentions exact lookup misses: 15.8 points of person recall@∞, 12.6 from mentions with no candidate and 3.1 from those retrieved with the wrong ones. It does so at a median of 51 candidates and a mean of 369 when the index is not restricted, which is a ranking problem handed downstream. The class restriction changes nothing about what is recovered but removes about 60% of the competitors (median 51 to 21, mean 369 to 138). The two-token rule is precise and recovers 8.5%. The person shortfall is therefore partly a limit of exact matching rather than of the resources' content: the surname is on record, the full form as written is not.

Events

Events have a rank-10 gap of 21.8 points, 9 to 24 times that of any other class: the correct event is retrieved but buried, because event mentions are generic noun phrases (□□□□□□□□□□□ ,□□□□□ ,□□□□□□□□) that match hundreds of entities. The better prior reduces the gap from 28.8 under popularity alone but does not close it. Events are also worst on absent-with-candidates (21.9%) and, with it, worst on total coverage failure (35.5%): often the matching entities do not include the one meant. Events fail on both counts, and the ranking half is the one a disambiguator could address.

Other classes

Places, organisations, works and the residual class have rank-10 gaps between 1.4 and 2.5 points against coverage failures between 18.1 and 27.3. At rank 1 the gaps are 14 to 20 points, comparable to coverage.

4.6 Out-of-domain check: Mewsli-9

Every result above is measured on mentions drawn from Arabic Wikipedia. Mewsli-9 (Botha et al., 2020) provides 7,367 Arabic mentions from 1,468 WikiNews articles linked to Wikidata, with a published alias-table baseline. We re-ran the six layers on it unchanged, with noal normalisation; 95.7% of its mentions target entities that have an Arabic Wikipedia article, and 71.8% are literal matches of that article's title. Table 9 gives the results.

Table 9
Mewsli-9 Arabic, noal normalisation, P(e|m) prior, all 7,367 mentions; the last two columns are the 2,075 non-literal ones (prior-independent quantities). Single run; literal and non-literal hit counts sum to the total at every layer.

Layer	Empty %	R@1	R@10	R@∞	Gap@10	Empty % (NL)	R@∞ (NL)
S0	13.98	75.00	78.23	78.58	0.35	48.00	32.34
S1	8.10	81.25	84.99	85.46	0.48	27.71	55.81
S2	7.89	81.50	85.27	85.75	0.48	27.37	56.29
S3	7.68	83.39	87.23	87.77	0.54	27.28	56.58
S4	5.25	86.96	91.19	91.87	0.68	18.65	71.13
S5	2.85	88.95	93.36	95.38	2.02	10.12	83.61

External corroboration

Our full stack reaches recall@1 of 88.95 and recall@10 of 93.36 on Mewsli-9 Arabic. Botha et al. (2020) report 0.89 and 0.93 for their alias table on the same mentions, which pools names from 104 languages where ours holds Arabic surface forms only. The resources assembled here have the reach of that table, and its ceiling, which they did not report, is 95.4.

The complete dictionary on news

Because Mewsli-9 is not drawn from Wikipedia, the anchor dictionary can be built from all 16 buckets without leakage. It raises recall@∞ from 95.38 to 95.85 over all mentions and from 83.61 to 85.25 on the non-literal ones, leaves recall@1 unchanged at 88.95, and lowers recall@10 by 0.2 as more candidates compete. The head-heavy news set is already well served by half the encyclopaedia. The level agrees with the benchmark once the benchmark is weighted by occurrence: 94.6 and 95.5 over all mentions at the two dictionary sizes (Section 4.2) against 95.38 and 95.85 here. The gap between the news set and the pair sample is the weighting, not the domain.

Difficulty relative to the benchmark

On non-literal mentions the ceiling is 83.6 on Mewsli-9 against 74.4 on our set. Mewsli-9 Arabic is head-heavy (2,232 distinct entities for 7,367 mentions, 71.8% literal), where the pair sample holds far more of the tail. Wikipedia-derived evaluation is often assumed to be optimistic relative to news; for candidate generation on Arabic the direction is the reverse, and the reason is entity frequency.

4.7 What the coverage failures are

The 25.6% of non-literal mentions with no correct candidate at S5 are a mixed population. We hand-classified a stratified sample of 299 of them, at least 30 per entity class and the remainder in proportion to class size, into 10 categories. The annotator saw the mention, the link target, and the Wikidata labels, description and aliases of the target. A second annotator independently labelled 100 of the 299 and agreed with the first on 81% (raw agreement). Because the sample is stratified, Table 10 reports the raw sample share and the share reweighted to the class distribution of the failure population; grouped figures in the text use the reweighted shares.

Table 10

Manual classification of 299 coverage failures at S5. *Sample* is the share of the 299 labelled cases; *reweighted* re-scales each class to its share of the 24,461 failures in the population, since the sample holds at least 30 cases per class.

Category	Meaning	Sample %	Reweighted %
variant	correct entity mention; a legitimate name form recorded in no resource	43.1	43.0
translit	transliteration variant of a foreign name	3.0	2.9
latin	mention written in Latin script	8.4	8.0
kberr	the resource holds a wrong Arabic name or a wrong sitelink for the entity	0.7	1.0
boundary	link span over-extends by one or more standalone function words (□□ □□□, …)	10.7	11.5
generic	common noun or generic term linked to a specific entity	8.4	9.9
context	reference resolvable only from the surrounding article	8.0	7.1
nisba	nisba adjective or demonym	1.3	1.5
linkerr	the gold link itself points to the wrong entity	7.7	8.5
other	none of the above	8.7	6.7

Grouping these answers the question that matters. 54.9% of coverage failures are genuine resource failures: the mention is a name a recogniser would emit, and no resource holds that form (or, in the kberr case, holds a wrong one). 30.0% are not entity mentions in the sense a recogniser produces (over-extended link spans, common nouns, contextual references and nisba adjectives) and would not arise on recogniser output. 8.5% are errors in the gold link, and 6.7% could not be classified. By class (Online Resource 1, Table S11), genuine resource failures are 73% of person and 72% of place failures, 60% of organisation, 52% of work, 46% of the residual class and 29% of event failures.

Two adjustments to the headline follow. In points of the non-literal set, the 25.6-point coverage failure is 14.0 points of genuine resource shortfall, 7.7 of benchmark artefact, 2.2 of gold error and 1.7 unclassified. Removing the artefacts and gold errors from the failures, the exact-lookup ceiling on entity mentions with correct links would be **82.6%** (84.2% without the unclassified residue). That is an upper bound, not an estimate. Only failures were classified, and a generic noun or an over-extended span that happens to match a key counts as a success by the same measure. If successes carry artefacts at the same rate, the ceiling on true entity mentions stays at 74.4%, and the value lies between. For persons and places 73% of the shortfall is real, which confirms Section 4.5. For events the scheme itself fails: 36.8% of event failures fit none of the 10 categories, because event mentions are often descriptive phrases (*the match between X and Y*) that no name resource could hold.

Remedies

Two of the categories suggested cheap remedies, and we tested both (Table 11). The first is a seventh normalisation operation that strips a standalone function word (□□ □□□ □□□□ □□□□ □□□ □□□□ …) from the start or end of the mention. The second adds the English Wikidata label of every entity in the pool as a further surface-form source, with case folding on both sides.

Table 11
Two remedies suggested by the manual analysis, and one further source, tested on the full stack, non-literal mentions, noal, P(e|m).

Configuration	Empty %	R@1	R@10	R@∞	Δ R@∞
S5	18.17	56.99	71.92	74.48	—
+ function-word stripping	18.01	57.06	72.02	74.62	+ 0.14
+ English labels	17.96	57.16	72.14	74.70	+ 0.22
+ both	17.79	57.23	72.24	74.84	+ 0.36
+ Egyptian Arabic labels (arz)	18.07	57.06	72.00	74.56	+ 0.08

The reference row differs from Table 4 by 0.04 because case folding is now applied to Latin strings on both sides. Neither remedy delivers what the categories implied, and the failure population says why. Spans that begin or end with a standalone function word are 1.1% of the failures (263 of 24,426) rather than the 11.5% of the boundary category; the seventh operation recovers 44% of those, hence 0.14 points. Spans that the Light10 prefix set changes and the noal rule does not (an attached □ or □□, or a proclitic before a bare noun) are 13.7% of the failures (3,338). Light10 recovers 37% of them, the 1.4 points of Section 4.4. These, and not standalone words, are most of what the annotator called a boundary error; the rest carry more than one extra word (□□□□□□□ □□□□□ □□ for □□□□□□□). Latin-script mentions are 6.9% of the failures (1,673), close to the category's 8.0%, but the English label matches only 12.6% of them (210). The rate is 27% for places, 20% for persons, 16% for works, 7% for organisations and the residual class, and 2% for events, and it falls the more often the pair is linked. A Latin-script mention in Arabic text is more often an abbreviation, a name in a third language, a partial name or a mark-up remnant than the exact English label. On Mewsli-9, whose mentions are all Arabic-script, both changes are null. The last row of Table 11 tests the largest untried Arabic-script source, the 1.9 million Egyptian Arabic labels of Online Resource 1 (Section S1), over S5. 0.08 points confirms that the layer set of Table 2 is not missing a material resource. The manual categories describe the failures correctly and mis-estimate their recoverability, which is a reason to test every remedy a manual analysis suggests.

Measured on the labelled cases themselves (Table 12), the remedies reproduce this split in miniature, with one exception. Case folding alone recovers 1 of the 299. Function-word stripping recovers 2 of the 32 boundary cases and the Light10 set 9 more, the attached-proclitic cases; the remaining 21 are recovered by neither rule. English labels recover 10 of the 25 Latin-script cases, or 40%, against 12.6% over all 1,673 Latin-script failures of the same population. Composition explains part of the difference, since the labelled cases hold more places and persons, where the label matches most often; but the population rates for the classes of the 25 cases predict 3.7 recoveries rather than 10. The rest we attribute to the small count and to the labelling itself. The annotator saw the English label while labelling, and a Latin-script mention that visibly matched it was the clearest case for the label *latin*; 25 of the 30 Latin-script cases in the sample received it. We take the population figure as the estimate. The n-gram matcher of Section 4.8, run on the same 299, separates the

categories by what they are. At 100 candidates it recovers 80% of the variants and 97% of the boundary cases, which are string deformations of a recorded name. It recovers 32% of the Latin-script and 44% of the transliteration cases, which cross scripts, 24% of the generic ones, 46% of the contextual references and 48% of the wrong gold links, the last being coincidences of spelling. Its recovery on the 299 (61.2% at 100) is 4 points below the population figure of Table 13 because the sample over-represents the classes that fail hardest.

Table 12
Recovery on the 299 labelled failures, by category: the share of each category's cases whose correct entity enters the candidate set under each remedy of Table 11 and under the Light10 prefix rule of Section 4.4 (exact lookup, cumulative on S5 with case folding), and within the top 10 and top 100 of the character n-gram matcher of Section 4.8. The two exact-match remedies together recover 14 cases.

Category	n	Case fold	+ function word	+ English labels	Light10	n-gram @10	n-gram @100
variant	129	0.0	0.8	0.0	3.9	60.5	79.8
translit	9	0.0	0.0	0.0	0.0	33.3	44.4
latin	25	4.0	4.0	40.0	0.0	28.0	32.0
kberr	2	0.0	0.0	0.0	0.0	100.0	100.0
boundary	32	0.0	6.2	0.0	28.1	84.4	96.9
generic	25	0.0	0.0	4.0	0.0	0.0	24.0
context	24	0.0	0.0	0.0	4.2	20.8	45.8
nisba	4	0.0	0.0	0.0	0.0	0.0	25.0
linkerr	23	0.0	0.0	0.0	0.0	17.4	47.8
other	26	0.0	0.0	0.0	0.0	15.4	23.1
all	299	0.3	1.3	3.7	5.0	43.5	61.2

4.8 Beyond exact match: what fuzzy and dense matching recover

The ceiling above is a property of exact lookup. To say how much of the shortfall is recoverable by string similarity, without context, we ran two matchers over the same benchmark and the same pool: every distinct normalised surface form of layer S5 (6,675,784 strings for 7,383,026 string–entity pairs). The lexical matcher represents each string by TF-IDF-weighted character 3- and 4-grams, dropping features that occur in more than 30,000 strings, and retrieves by cosine similarity. The dense matcher embeds each string with multilingual-e5-small (Wang et al., 2024), a 384-dimensional multilingual sentence encoder used without fine-tuning, and retrieves by cosine similarity over the embeddings. In both, each non-literal mention retrieves its 200 nearest strings, which are mapped to their entities and truncated at 100 distinct candidates. Table 13 reports recall over all non-literal mentions, where the matcher replaces exact lookup, and over the mentions that fail at S5, where it supplements it.

Table 13

String matching on the benchmark, non-literal mentions. *Alone*: the matcher's own recall over all 95,699 mentions. *On S5 failures*: recall over the 24,458 mentions whose correct entity is absent at S5 (the manual sample of Section 4.7 was drawn from an earlier run of the same grid that counted 24,461; the two differ by three mentions). *Composite*: exact lookup at S5 plus the matcher's top-*k* on its failures.

Configuration	n	R@1	R@10	R@50	R@100
character n-gram, alone	95,699	42.51	76.02	87.07	89.86
dense (multilingual-e5-small), alone	95,699	52.28	80.97	89.81	91.95
character n-gram on S5 failures	24,458	23.82	46.81	60.81	65.94
dense on S5 failures	24,458	33.19	55.11	66.94	71.60
composite: exact S5 + n-gram top-*k*	95,699	—	86.40	89.98	91.29
composite: exact S5 + dense top-*k*	95,699	—	88.53	91.55	92.74

Most of the exact-match shortfall is within reach of a string matcher that sees nothing but the mention. With 100 candidates per mention, the n-gram matcher recovers 65.9% of the failures and the dense matcher 71.6%, and the composite ceiling on non-literal mentions rises from 74.4 to 91.3 and 92.7. Either matcher alone, with no exact lookup at all, retrieves the entity within 10 candidates for more mentions than the exact-match ceiling admits at any depth (76.0 and 81.0 against 74.4), and within 100 for 89.9 and 92.0. What they cost is candidates, 100 per mention against a median of 2 for exact lookup, and rank 1, where the dense matcher (52.3) is below the dictionary with its prior (57.0). They are recall devices to be placed in front of a ranker rather than replacements for the dictionary. The dense matcher's 5-point advantage over n-grams on the failures is where it reads across scripts and transliterations that share few character n-grams. The 7 points that remain beyond 100 dense candidates are the mentions of Section 4.7 that no string resemblance reaches: descriptive event phrases, contextual references and wrong gold links. A contextual bi-encoder (Botha et al., 2020; Wu et al., 2020), which also sees the sentence, is what would bound them from above.

5. Discussion

What the ceiling means

Exact-match candidate generation over every public Arabic surface-form resource we could assemble, with a dictionary from half of Arabic Wikipedia, leaves 25.6% of non-trivial mentions without the correct entity on the pair sample. Weighted by frequency in text the figure is 13.2%. Of the 25.6 points, 14.0 are genuine resource gaps once artefacts and gold errors are set aside (Section 4.7), for a ceiling of at most 83% on true entity mentions. The complete anchor text of Arabic Wikipedia would add 2 to 4 points, depending on the sample (Section 4.2), and no more; the 2.55 points of rank-10 failure are all a better re-ranker could recover. Three routes go beyond it. String similarity is the cheapest. With 100 candidates per mention, a character n-gram matcher recovers 66% of the exact-match failures and a dense string encoder 72% (Section 4.8). The practical ceiling of a dictionary-plus-matcher generator is therefore about 93%, and the last 7 points are mentions no string resemblance reaches. Cross-lingual retrieval (dense retrieval, Wu et al., 2020; the multilingual dual encoder of Botha et al., 2020; generative linking, De Cao et al., 2022) matches a mention against

representations that need not be Arabic. On Mewsli-9 Arabic the dual encoder reaches recall@10 of 0.98 where the alias table stops at 0.93. The third is annotation: Arabic names for the entities that lack them, which no retrieval method can conjure.

For system builders

The conditional in Section 4.2 is the actionable result. With Arabic Wikipedia anchor text, further resources add about 2 points; without it, redirects are the strongest substitute and Wikidata aliases the next, and article titles and ParaNames are near-redundant with the label in every configuration. Strip the definite article: it raises the ceiling at every layer and, under a mention-conditioned prior, rank-1 accuracy as well. English labels and function-word stripping move the ceiling by 0.1 to 0.4 points (Section 4.7). And read the anchor dictionary's gain with the unseen-pair figure beside it: most of it comes from pairs recorded in its own half, and the unseen figure bounds what it delivers on the rest.

For resource curation

The 14.2% alias rate and the 1.2 labels per article of Table 3 are the most actionable numbers in the paper: Arabic names in Wikidata stop where Arabic articles stop. Arabic Wikipedia redirects, which record the same kind of variation as aliases, exist for 51% of Arabic articles. Importing redirect targets as aliases is mechanical and would close much of the gap for entities that have Arabic articles, though nothing for the 82% of English-covered entities with no Arabic name at all, which needs annotation rather than transformation. The 460,644 items whose only Arabic-tagged name is a Latin-script surname (Section 4.1) and the dialect labels of Online Resource 1 (Section S1) are worth a cleanup pass. Per-language Wikidata counts should be filtered for entity type, script and duplication before they are quoted, since bots are the dominant label editors (Kaffee et al., 2017).

Where the bottleneck sits

Whether entity linking is a disambiguation problem depends on the rank one cares about. A system that re-ranks 10 candidates loses 10 times more to coverage than to ranking in every class but events; a bare lookup system loses to ranking about as much as to coverage. The prior matters too. The popularity prior made the anchor dictionary look harmful to easy cases and events look worse than they are, and both effects shrink under the mention-conditioned prior standard since Milne and Witten (2008). Studies that report only recall@1 under a popularity prior will misattribute ranker weakness to the resources; the three-way decomposition costs two extra columns and changes what a result means.

6. Limitations

Domain

We assumed, following Fu et al. (2020), that Wikipedia-derived mentions would make the ceiling optimistic for news. Section 4.6 shows the opposite on Mewsli-9, and Section 4.2 shows why: once the benchmark is weighted by occurrence its all-mention ceiling matches Mewsli-9's within a point. Text from other domains may sit above or below our figures depending on how many rare entities it names; social media and dialectal text remain untested.

Matchers without context

Both matchers of Section 4.8 see the mention string only, so their recovery rates are lower bounds on what a contextual dense retriever would reach. Their 100 candidates per mention are a cost we did not trade off against a ranker, and the dense encoder was used as released, without fine-tuning.

Two samplings

The pair sample of Section 3.2 draws each mention–entity pair at most once; every table in Section 4 is on it, and it over-weights rare pairs relative to text. The occurrence sample of Section 4.2 gives the ceiling text would see, 86.8 at the same setting, with a standard deviation of 0.3 across held-out buckets. The choice of weighting moves the non-literal ceiling by 12 points and is therefore what limits the precision of the headline. The pair figure remains the headline because the decomposition, the ablations and the manual analysis were done on that sample; the figure for running text is 86.8 ± 0.3, and 94.6 over all mentions.

Manual analysis

299 classified failures give overall shares to about ± 5 points and per-class shares to ± 10–15. The second-annotator check covers 100 of them with raw agreement only, and the 10-category scheme left 37% of event failures unclassified.

Reference languages

Table 3 places Arabic's coverage beside other languages, but the candidate ceiling was measured for Arabic alone. The layered design is language-independent and the released tables make the comparison possible.

Gold entities

All gold entities have Arabic Wikipedia articles, so oracle coverage on the benchmark is trivially complete by construction (Section 3.2). Section 4.1 answers the coverage question on an external denominator; the two measurements are not over the same mentions, and the 82% of entities without an Arabic name are invisible to every recall figure in the paper.

Anchors that are not entity mentions

Nisba adjectives and common nouns linked to an entity, which a recogniser would not emit, fall mostly into the residual *other* class (38.6% of non-literal mentions). Claims here are about candidate generation over linkable spans.

The article rule

The fuller Light10 set raises the S5 ceiling by 1.4 points and a bare-proclitic variant by 2.3 (Section 4.4). The full grid was not recomputed under them, so the headline is conservative by that margin, and a morphological segmenter was not tested.

The prior

Candidates are ranked by bucket-0 link frequency, so recall@k at Wikidata-only layers is not a Wikidata-only quantity; recall@∞ and the empty rate are unaffected and carry the ceiling claims.

Snapshots

Arabic Wikipedia dumps are those of 1 September 2026 (pages-articles retrieved 18 September). Wikidata was read through the QLever public endpoint on 18–25 September 2026, whose index build date we do not control; live-API checks were made on 25 September 2026. ParaNames is release v2024.05.07.0, so its 95.4% overlap with Wikidata labels compares a 2024 snapshot with a 2026 one.

7. Conclusion

Arabic entity linking is bounded before disambiguation begins, and the bound now has a shape rather than a number. 82% of the entities English Wikipedia covers have no Arabic name in Wikidata, a share Persian and Turkish share, and 85.8% of the entities Arabic Wikipedia covers have exactly one. For the entities that do have an Arabic article, exact-match lookup over the public Arabic surface-form resources places the correct entity in the candidate set for 74.4% of non-trivial mentions on a sample that draws each pair once. Weighted by occurrence, the figure is 86.8%. The shortfall at rank 10 is coverage rather than ranking. The ceiling is a function of dictionary size, the seen-pair share explains every figure exactly, and the split itself moves it by 0.3 points. String matching without context recovers most of what exact lookup misses at 100 candidates per mention, for a composite ceiling near 93%; on Mewsli-9 Arabic the same resources match the published alias-table baseline.

Which resources matter depends on what is already present. Anchor text from Arabic Wikipedia reaches 72.3% on its own and leaves 2 points for everything else. Without it, redirects are the best substitute and Wikidata aliases the next, while resources that restate an entity's canonical name add almost nothing. The anchor dictionary's gain is almost entirely on pairs it has recorded; on the rest the whole stack reaches 12%. Stripping the Arabic definite article is the most valuable normalisation step under both cumulative and leave-one-out ablation, and it still pays at rank 1 under a mention-conditioned prior.

Failure is not uniform. Persons fail on name forms that no resource records exactly, 51% of which surname lookup would recover at a heavy ambiguity cost. Events, the class with the highest coverage failure, are also the only one where the retrieved candidates cannot be ranked; the other classes fail on coverage by a wide margin at rank 10. Reported coverage cannot be taken at face value either. Wikidata's 351,466 Algerian Arabic labels reduce to 538 once labels not in Arabic script, labels identical to the Modern Standard Arabic one, and 113,669 calendar dates from one automated import are removed.

The benchmark, the anchor dictionary, the surface-form pool and the analysis code are released (Mahammedi et al., 2026). These figures can then be recomputed on later snapshots, and the same layered, order-independent decomposition applied to other languages whose Wikidata coverage looks better on paper than it is.

Declarations

Funding. No external funding was received for this work.

Competing interests. The authors declare no competing interests.

Research involving human participants and/or animals. Not applicable. The study analyses publicly available resources (Wikidata, Wikipedia and published corpora) and involved no human participants or animals.

Informed consent. Not applicable.

Data availability. The evaluation set, anchor dictionary, surface-form pool, component snapshots and dialect label tables are available from Zenodo at https://doi.org/10.5281/zenodo.22964063 under CC BY-SA 4.0 (Mahammedi et al., 2026). Raw Wikipedia dumps are not redistributed.

Code availability. The analysis code is released in the same Zenodo record (Mahammedi et al., 2026) as code.zip under the MIT licence. It holds the notebooks that produce every table and figure, in pipeline order, and a script that reproduces the figures from the saved tables.

Supplementary information. Online Resource 1 (ESM_1.pdf) contains the dialect label audit (Section S1), the audit of the ANERD corpus (Section S2) and the implementation details needed for reimplementation (Section S3). It also holds the full result grids and the rotation table (Section S4) and the tables referenced from the article (Section S5).

Author contributions. A.A.M.: conceptualisation, methodology, software, data curation, formal analysis, visualisation, writing (original draft). A.B. and Y.Y.: methodology, validation, writing (review and editing). D.B.: supervision, writing (review and editing). All authors read and approved the manuscript.

Use of AI tools. The authors used a large language model (Gemini, Google) to assist with drafting and editing the manuscript text.

References

1. Alotaibi, M. S., & Menai, M. E. B. (2026a). ANERD: A large-scale Arabic corpus for named entity recognition and disambiguation. *Data*, *11*(9). https://doi.org/10.3390/data11090243. Article 243.
2. Alotaibi, M. S., & Menai, M. E. B. (2026b). *ANERD: A large-scale Arabic corpus for named entity recognition and disambiguation* (Version 1.0.0) [Data set]. Zenodo. https://doi.org/10.5281/zenodo.21909063
3. Ayedh, A., Tan, G., Alwesabi, K., & Rajeh, H. (2016). The effect of preprocessing on Arabic document categorization. *Algorithms*, *9*(2). https://doi.org/10.3390/a9020027. Article 27.
4. Bast, H., & Buchhold, B. (2017). QLever: A query engine for efficient SPARQL+Text search. In *Proceedings of the 2017 ACM on Conference on Information and Knowledge Management* (pp. 647–656). Association for Computing Machinery. https://doi.org/10.1145/3132847.3132921
5. Benajiba, Y., Rosso, P., & Benedí Ruiz, J. M. (2007). ANERsys: An Arabic named entity recognition system based on maximum entropy. In *Computational Linguistics and Intelligent Text Processing: 8th International Conference, CICLing 2007* (Lecture Notes in Computer Science, Vol. 4394, pp. 143–153). Springer. https://doi.org/10.1007/978-3-540-70939-8_13

6. Botha, J. A., Shan, Z., & Gillick, D. (2020). Entity linking in 100 languages. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)* (pp. 7833–7845). Association for Computational Linguistics. https://doi.org/10.18653/v1/2020.emnlp-main.630
7. Bouguettoucha, H. H., & Djouablia, I. (2025). *ELNER-DZ: A dataset for named entity recognition and entity linking in Algerian Arabic dialect* (Version 1) [Data set]. Zenodo. https://doi.org/10.5281/zenodo.15794817
8. Bunescu, R., & Paşca, M. (2006). Using encyclopedic knowledge for named entity disambiguation. In *Proceedings of the 11th Conference of the European Chapter of the Association for Computational Linguistics* (pp. 9–16). Association for Computational Linguistics. https://aclanthology.org/E06-1002/.
9. Cucerzan, S. (2007). 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)* (pp. 708–716). Association for Computational Linguistics. https://aclanthology.org/D07-1074/.
10. Dahou, A. H., & Cheragui, M. A. (2023). DzNER: A large Algerian named entity recognition dataset. *Natural Language Processing Journal*, *3*., Article 100005. https://doi.org/10.1016/j.nlp.2023.100005
11. De Cao, N., Wu, L., Popat, K., Artetxe, M., Goyal, N., Plekhanov, M., Zettlemoyer, L., Cancedda, N., Riedel, S., & Petroni, F. (2022). Multilingual autoregressive entity linking. *Transactions of the Association for Computational Linguistics*, *10*, 274–290. https://doi.org/10.1162/tacl_a_00460
12. El Kholy, A., & Habash, N. (2012). Orthographic and morphological processing for English–Arabic statistical machine translation. *Machine Translation*, *26*(1–2), 25–45. https://doi.org/10.1007/s10590-011-9110-0
13. Fu, X., Shi, W., Yu, X., Zhao, Z., & Roth, D. (2020). Design challenges in low-resource cross-lingual entity linking. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)* (pp. 6418–6432). Association for Computational Linguistics. https://doi.org/10.18653/v1/2020.emnlp-main.521
14. Gad-Elrab, M. H., Yosef, M. A., & Weikum, G. (2015a). EDRAK: Entity-centric data resource for Arabic knowledge. In *Proceedings of the Second Workshop on Arabic Natural Language Processing* (pp. 191–200). Association for Computational Linguistics. https://doi.org/10.18653/v1/W15-3224
15. Gad-Elrab, M. H., Yosef, M. A., & Weikum, G. (2015b). Named entity disambiguation for resource-poor languages. In *Proceedings of the Eighth Workshop on Exploiting Semantic Annotations in Information Retrieval* (pp. 29–34). Association for Computing Machinery. https://doi.org/10.1145/2810133.2810138
16. Hachey, B., Radford, W., Nothman, J., Honnibal, M., & Curran, J. R. (2013). Evaluating entity linking with Wikipedia. *Artificial Intelligence*, *194*, 130–150. https://doi.org/10.1016/j.artint.2012.04.005
17. Hamad, N., Khalilia, M., & Jarrar, M. (2025). Konooz: Multi-domain multi-dialect corpus for named entity recognition. In *Findings of the Association for Computational Linguistics: ACL 2025* (pp. 7316–7331). Association for Computational Linguistics. https://doi.org/10.18653/v1/2025.findings-acl.382
18. Hoffart, J., Yosef, M. A., Bordino, I., Fürstenau, H., Pinkal, M., Spaniol, M., Taneva, B., Thater, S., & Weikum, G. (2011). Robust disambiguation of named entities in text. In *Proceedings of the 2011 Conference on Empirical Methods in Natural Language Processing* (pp. 782–792). Association for Computational Linguistics. https://aclanthology.org/D11-1072/.

19. Jarrar, M., Abdul-Mageed, M., Khalilia, M., Talafha, B., Elmadany, A., Hamad, N., & Omar, A. (2023). WojoodNER 2023: The first Arabic named entity recognition shared task. In *Proceedings of ArabicNLP 2023* (pp. 748–758). Association for Computational Linguistics. https://doi.org/10.18653/v1/2023.arabicnlp-1.83
20. Jarrar, M., Khalilia, M., & Ghanem, S. (2022). Wojood: Nested Arabic named entity corpus and recognition using BERT. In *Proceedings of the Thirteenth Language Resources and Evaluation Conference* (pp. 3626–3636). European Language Resources Association. https://aclanthology.org/2022.lrec-1.387/.
21. Kaffee, L. A., Piscopo, A., Vougiouklis, P., Simperl, E., Carr, L., & Pintscher, L. (2017). A glimpse into Babel: An analysis of multilinguality in Wikidata. In *Proceedings of the 13th International Symposium on Open Collaboration* (Article 14, pp. 1–5). Association for Computing Machinery. https://doi.org/10.1145/3125433.3125465
22. Kubeša, D., & Straka, M. (2023). *DaMuEL: A large multilingual dataset for entity linking* (arXiv:2306.09288). arXiv. https://doi.org/10.48550/arXiv.2306.09288
23. Larkey, L. S., Ballesteros, L., & Connell, M. E. (2002). Improving stemming for Arabic information retrieval: Light stemming and co-occurrence analysis. In *Proceedings of the 25th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval* (pp. 275–282). Association for Computing Machinery. https://doi.org/10.1145/564376.564425
24. Larkey, L. S., Ballesteros, L., & Connell, M. E. (2007). Light stemming for Arabic information retrieval. In A. Soudi, A. van den Bosch, & G. Neumann (Eds.), *Arabic computational morphology* (Text, Speech and Language Technology, Vol. 38, pp. 221–243). Springer. https://doi.org/10.1007/978-1-4020-6046-5_12
25. Mahammedi, A. A., Bouziane, A., Yahiaoui, Y., & Bouchiha, D. (2026). *Arabic entity linking surface-form coverage: Benchmark, anchor dictionary, surface-form pool and analysis code* (Version 1.1.0) [Data set and software]. Zenodo. https://doi.org/10.5281/zenodo.22964063
26. Marton, Y., Habash, N., & Rambow, O. (2013). Dependency parsing of Modern Standard Arabic with lexical and inflectional features. *Computational Linguistics*, *39*(1), 161–194. https://doi.org/10.1162/COLI_a_00138
27. Milne, D., & Witten, I. H. (2008). Learning to link with Wikipedia. In *Proceedings of the 17th ACM Conference on Information and Knowledge Management* (pp. 509–518). Association for Computing Machinery. https://doi.org/10.1145/1458082.1458150
28. Mohit, B., Schneider, N., Bhowmick, R., Oflazer, K., & Smith, N. A. (2012). Recall-oriented learning of named entities in Arabic Wikipedia. In *Proceedings of the 13th Conference of the European Chapter of the Association for Computational Linguistics* (pp. 162–173). Association for Computational Linguistics. https://aclanthology.org/E12-1017/.
29. Moussa, H. N., & Mourhir, A. (2023). DarNERcorp: An annotated named entity recognition dataset in the Moroccan dialect. *Data in Brief*, *48*., Article 109234. https://doi.org/10.1016/j.dib.2023.109234
30. Nounou, J., & Sabty, C. (2026). ArabNED: A novel dataset and evaluation of fine-tuned vs. zero-shot models for Arabic named entity disambiguation. *Procedia Computer Science*, *275*, 309–316. https://doi.org/10.1016/j.procs.2026.01.038
31. Obeid, O., Zalmout, N., Khalifa, S., Taji, D., Oudah, M., Alhafni, B., Inoue, G., Eryani, F., Erdmann, A., & Habash, N. (2020). CAMeL Tools: An open source Python toolkit for Arabic natural language processing. In

Proceedings of the Twelfth Language Resources and Evaluation Conference (pp. 7022–7032). European Language Resources Association. https://aclanthology.org/2020.lrec-1.868/.

32. Orlando, R., Cabot, H., Barba, P. L., E., & Navigli, R. (2024). ReLiK: Retrieve and LinK, fast and accurate entity linking and relation extraction on an academic budget. In *Findings of the Association for Computational Linguistics: ACL 2024* (pp. 14114–14132). Association for Computational Linguistics. https://doi.org/10.18653/v1/2024.findings-acl.839
33. Pan, X., Zhang, B., May, J., Nothman, J., Knight, K., & Ji, H. (2017). Cross-lingual name tagging and linking for 282 languages. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)* (pp. 1946–1958). Association for Computational Linguistics. https://doi.org/10.18653/v1/P17-1178
34. Ruder, S., Constant, N., Botha, J., Siddhant, A., Firat, O., Fu, J., Liu, P., Hu, J., Garrette, D., Neubig, G., & Johnson, M. (2021). XTREME-R: Towards more challenging and nuanced multilingual evaluation. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing* (pp. 10215–10245). Association for Computational Linguistics. https://doi.org/10.18653/v1/2021.emnlp-main.802
35. Sälevä, J., & Lignos, C. (2024). ParaNames 1.0: Creating an entity name corpus for 400 + languages using Wikidata. In *Proceedings of the 2024 Joint International Conference on Computational Linguistics, Language Resources and Evaluation (LREC-COLING 2024)* (pp. 12599–12610). ELRA and ICCL. https://aclanthology.org/2024.lrec-main.1103/
36. Shaalan, K. (2014). A survey of Arabic named entity recognition and classification. *Computational Linguistics*, *40*(2), 469–510. https://doi.org/10.1162/COLI_a_00178
37. Shen, W., Wang, J., & Han, J. (2015). Entity linking with a knowledge base: Issues, techniques, and solutions. *IEEE Transactions on Knowledge and Data Engineering*, *27*(2), 443–460. https://doi.org/10.1109/TKDE.2014.2327028
38. Spitkovsky, V. I., & Chang, A. X. (2012). A cross-lingual dictionary for English Wikipedia concepts. In *Proceedings of the Eighth International Conference on Language Resources and Evaluation (LREC'12)* (pp. 3168–3175). European Language Resources Association. https://aclanthology.org/L12-1109/.
39. Tsai, C. T., & Roth, D. (2016). Cross-lingual wikification using multilingual embeddings. In *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies* (pp. 589–598). Association for Computational Linguistics. https://doi.org/10.18653/v1/N16-1072
40. Vrandečić, D., & Krötzsch, M. (2014). Wikidata: A free collaborative knowledgebase. *Communications of the ACM*, *57*(10), 78–85. https://doi.org/10.1145/2629489
41. Wang, L., Yang, N., Huang, X., Yang, L., Majumder, R., & Wei, F. (2024). *Multilingual E5 text embeddings: A technical report* (arXiv:2402.05672). arXiv. https://doi.org/10.48550/arXiv.2402.05672
42. Wu, L., Petroni, F., Josifoski, M., Riedel, S., & Zettlemoyer, L. (2020). Scalable zero-shot entity linking with dense entity retrieval. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)* (pp. 6397–6407). Association for Computational Linguistics. https://doi.org/10.18653/v1/2020.emnlp-main.519
43. Yosef, M. A., Spaniol, M., & Weikum, G. (2014). AIDArabic: A named-entity disambiguation framework for Arabic text. In *Proceedings of the EMNLP 2014 Workshop on Arabic Natural Language Processing (ANLP)*

(pp. 187–195). Association for Computational Linguistics. https://doi.org/10.3115/v1/W14-3626

44. Zhou, S., Rijhwani, S., Wieting, J., Carbonell, J., & Neubig, G. (2020). Improving candidate generation for low-resource cross-lingual entity linking. *Transactions of the Association for Computational Linguistics*, *8*, 109–124. https://doi.org/10.1162/tacl_a_00303

Figures

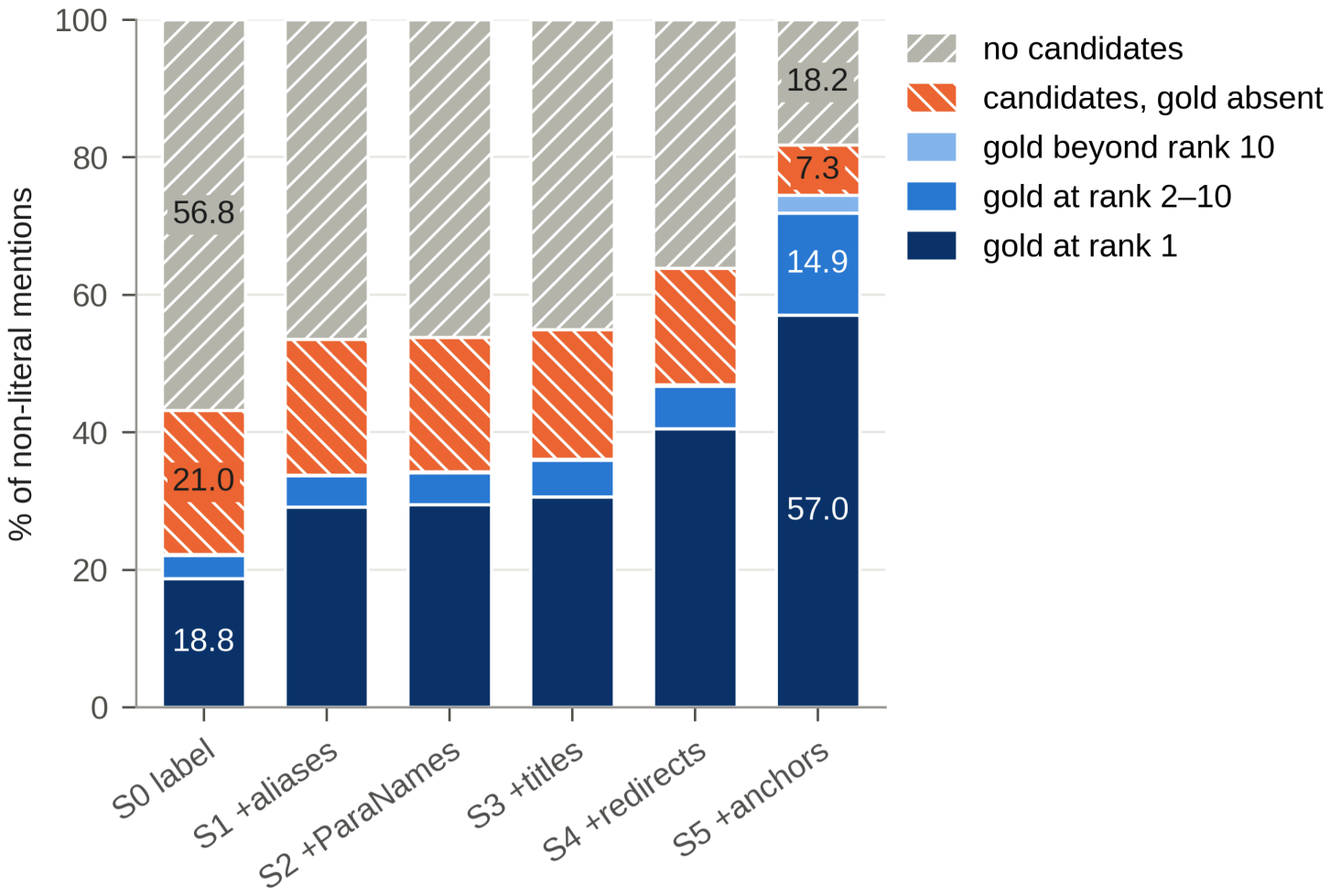


Figure 1

Failure decomposition by resource layer (non-literal mentions, noal, P(e|m))

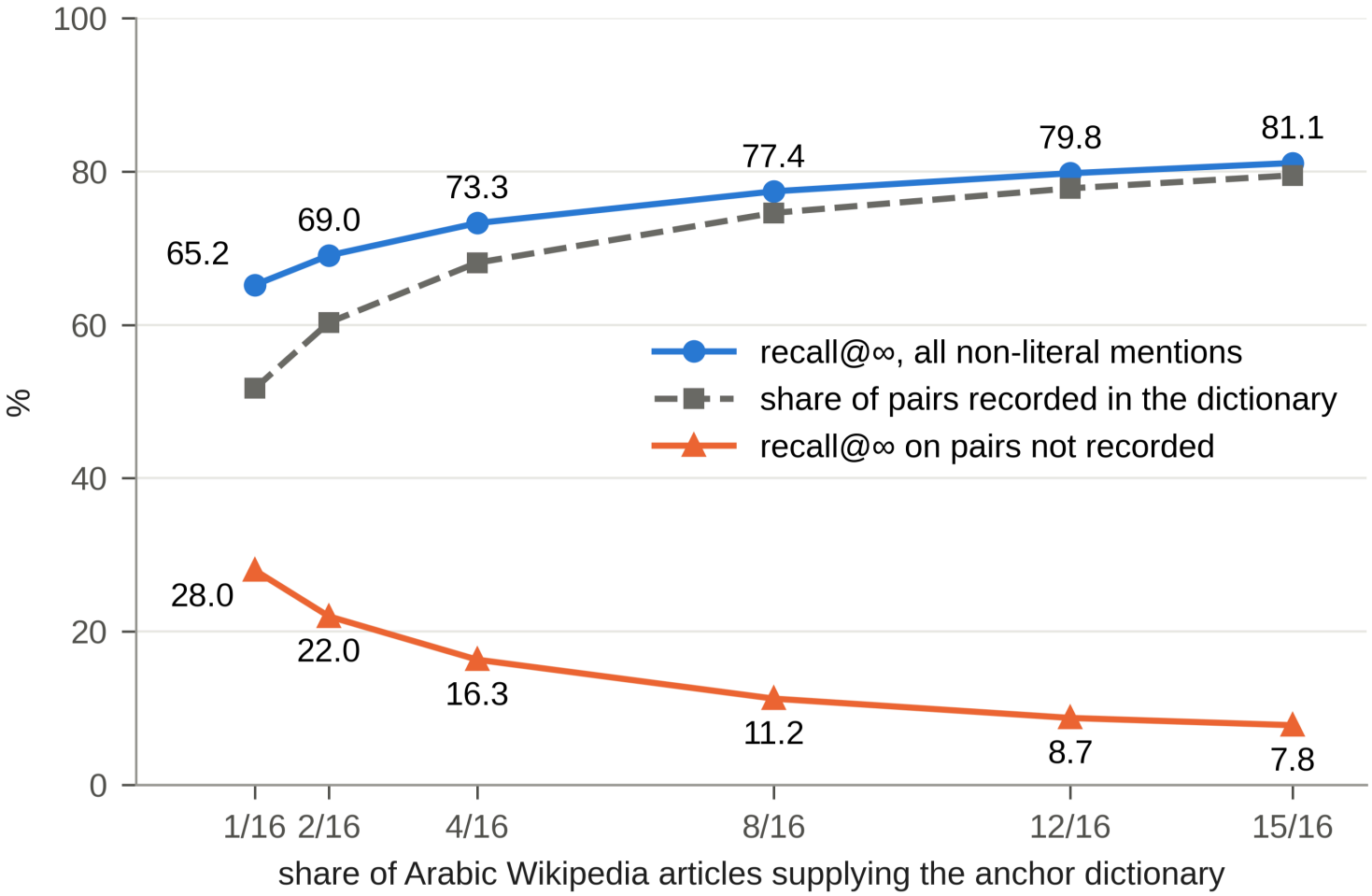


Figure 2

Ceiling, seen-pair share and unseen-pair ceiling by dictionary size (bucket-15 sample, non-literal mentions)

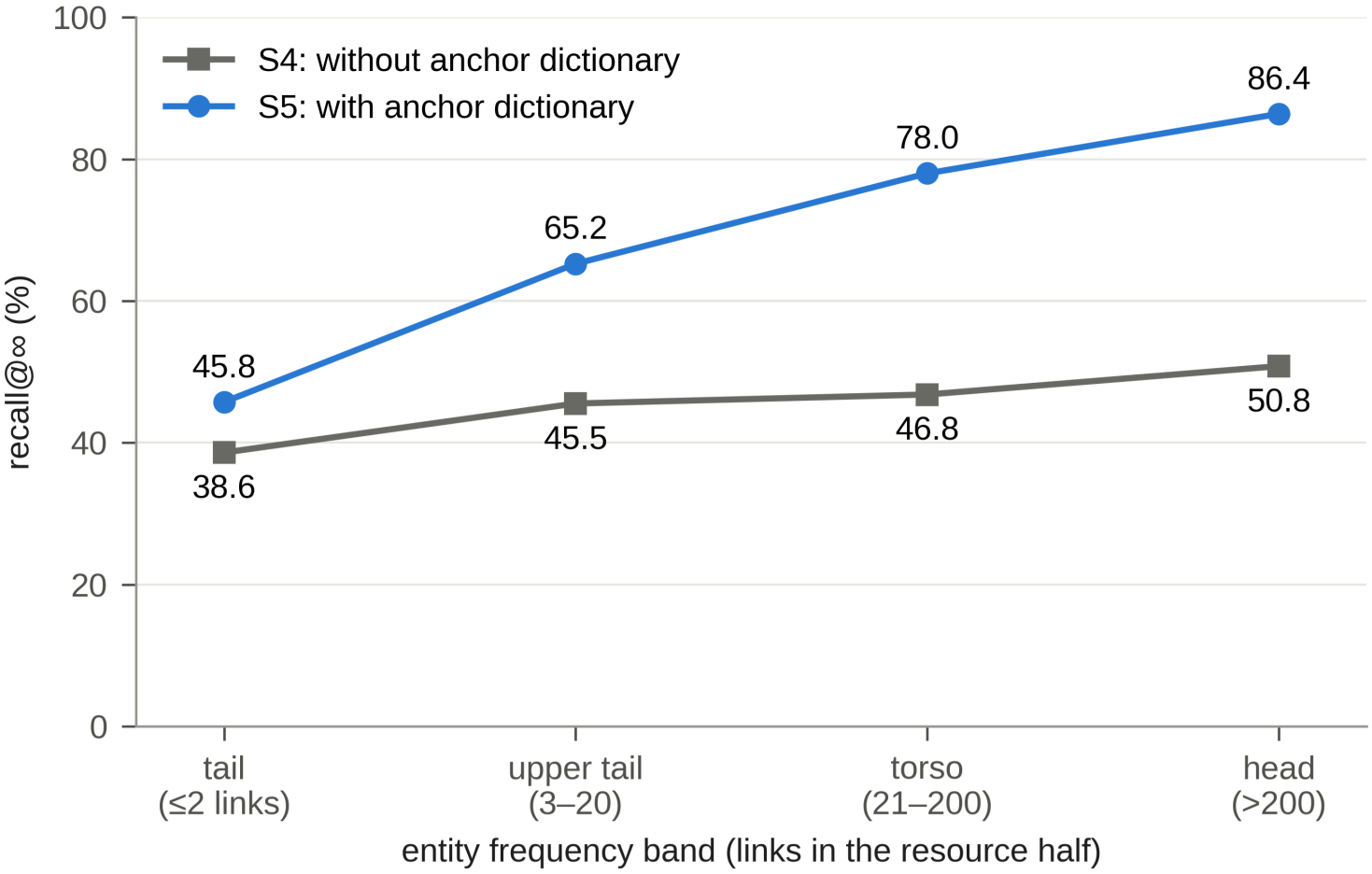


Figure 3

Candidate ceiling (recall@∞) by entity frequency band, non-literal mentions, with and without the anchor dictionary

Supplementary Files

This is a list of supplementary files associated with this preprint. Click to download.

- ESM1.docx