A Appendix

Table 3: Generated Datasets in terms of number of entities $|\mathcal{V}|$, triples $|\mathcal{E}|$, ground-truth summaries, density as $|\mathcal{E}|/\binom{|\mathcal{V}|}{2}$, graph connectivity, number of components, sampling method to select the entities and the subgraph, minimum and maximum node degree and, running time.

(a) Small Datasets

Metric	WikiLitArt	WikiCinema	WikiPro	WikiProFem
Entities ($\|\mathcal{V}\|$)	85 346	70 753	79 825	79 926
Relations ($\|\mathcal{E}\|$)	136 950	126 915	125 912	123 193
Target Entities	494	493	493	468
Density	0.000018	0.000018	0.000019	0.000019
Sampling method	Random Walk	Random Walk	Random Walk	Random Walk
Connected-graph	Yes	Yes	Yes	Yes
Num-comp	1	1	1	1
Min Degree	1	1	1	1
Max Degree	2172	3005	2060	3142
Run-time (seconds)	91.934	118.014	126.119	177.63

(b) Medium Datasets

Metric	WikiLitArt	WikiCinema	WikiPro	WikiProFem
Entities ($\|\mathcal{V}\|$)	128 061	101 529	119 305	122 728
Relations ($\|\mathcal{E}\|$)	220 263	196 061	198 663	196 838
Target Entities	494	493	493	468
Density	0.000013	0.000019	0.000014	0.000013
Sampling method	Random Walk	Random Walk	Random Walk	Random Walk
Connected-graph	Yes	Yes	Yes	Yes
Num-comp	1	1	1	1
Min Degree	1	1	1	1
Max Degree	3726	5124	3445	5282
Run-time (seconds)	155.36	196.413	208.157	301.718

(c) Large Datasets

Metric	WikiLitArt	WikiCinema	WikiPro	WikiProFem
Entities ($\|\mathcal{V}\|$)	239 491	185 098	230 442	248 012
Relations ($\|\mathcal{E}\|$)	466 905	397 546	412 766	413 895
Target Entities	494	493	493	468
Density	0.000008	0.00001	0.000008	0.000007
Sampling method	Random Walk	Random Walk	Random Walk	Random Walk
Connected-graph	Yes	Yes	Yes	Yes
Num-comp	1	1	1	1
Min Degree	1	1	1	1
Max Degree	8599	12189	7741	12939
Run-time (seconds)	353.113	475.679	489.409	768.99

A.1 Additional WIKES details

- **Dataset and Metadata:** The dataset is available at `https://github.com/msorkhpar/wiki-entity-summarization`. We generate four datasets in three sizes: small, medium, and large. Each size has an entire graph that includes all seed nodes and train-test-validation splits. In Table 9, we provide information on the proportion of seed nodes in each of the datasets. Moreover, Table 3 provides detailed information such as the number of entities $|\mathcal{V}|$, triples $|\mathcal{E}|$, ground-truth summaries, density, graph connectivity, number of components, sampling method used to select the

entities and the subgraph, minimum and maximum node degree, and running time for each of the datasets. Moreover, metadata is also in the same github repository.

- **WIKES toolkits:** We offer a comprehensive toolkit designed to facilitate working with our datasets. This toolkit includes features for downloading, loading, and manipulating pre-generated graph datasets. You can access our toolkit at `https://pypi.org/project/wikes-toolkit/`

- **Dataset Formats:** We generate our dataset in CSV format. The entity files are formatted according to Table 4. We also provide files containing target entities and their categories, as detailed in Table 8. The predicate files, described in Table 6, contain predicates along with their corresponding labels and descriptions. The triple file, presented in Table 7, includes the subject, object, and predicate IDs of the nodes (Wikidata items) and edges (Wikidata predicates). The ground-truth file, shown in Table 8, contains the subject, object, and predicate. Moreover, we provide the graph version of our dataset in GraphML and PKL formats in our release `https://github.com/msorkhpar/wiki-entity-summarization/releases/tag/1.0.5`.

- **URL to metadata record:** Since we have several datasets, in our GitHub Repository, we provide the Croissant URL metadata. The metadata for WikiLitArt small dataset is `https://github.com/msorkhpar/wiki-entity-summarization/releases/download/1.0.5/WikiLitArt-m.json`. The metadata format is consistent across all datasets. The metadata files are included alongside the datasets in the GitHub release.

- **Preprocessing URL:** You can find our preprocessing code for cleaning and preparing Wikipedia and Wikidata at the following link: `https://github.com/msorkhpar/wiki-entity-summarization-preprocessor`. Access and loading of preprocessed dump:

  - Neo4j database: `https://github.com/msorkhpar/wiki-entity-summarization-preprocessor/releases/tag/Neo4j-1.0.0`.
  - PostgreSQL: `https://github.com/msorkhpar/wiki-entity-summarization-preprocessor/releases/tag/PostgreSQL-1.0.0`.

- **Authors responsibility statement and License:** The authors are held responsible for copyright infringement, but assume no responsibility or liability for any misuse of the data. This project is licensed under the CC BY 4.0 License. See here `https://github.com/msorkhpar/wiki-entity-summarization/blob/main/LICENSE`

- **WIKES Generator Code:** The code for running the WIKES generator is available in the GitHub repository at `https://github.com/msorkhpar/wiki-entity-summarization`. The code allows to generate the same datasets as those provided in the paper or to create your own custom datasets.

- **Maintenance and Long Term Preservation** The authors of WIKES are dedicated to the ongoing maintenance and preservation of this dataset. This includes tracking and resolving issues identified by the community post-release. We will closely monitor user feedback through the GitHub issue tracker. The data is hosted on GitHub, ensuring reliable and stable storage.

- **Intended users:** The intended users are NLP and knowledge graph researchers who wish to generate summaries using the textual information of the entities (nodes) in knowledge graphs. The suitable use case for this dataset is evaluating entity summarization models to determine their ability to detect summaries accurately.

Field	Description	Datatype
id	Incremental integer starting by zero	int
entity	Wikidata qid, e.g. 'Q76'	string
wikidata_label	Wikidata label (nullable)	string
wikidata_desc	Wikidata description (nullable)	string
wikipedia_title	Wikipedia title (nullable)	string
wikipedia_id	Wikipedia page id (nullable)	long

Table 4: `{variant}-{size}-{dataset_type}-entities.csv` file contains entities. An entity is a Wikidata item (node) in our dataset. variant_index refers to the dataset id (detailed information is in our Github).

Field	Description	Datatype
entity	id key in Table 4	int
category	category	string

Table 5: `{variant}-{size}-{dataset_type}-root-entities.csv` contains root entities. A root entity is a seed node described previously. variant_index refers to the dataset id (detailed infomation is in our Github).

Field	Description	Datatype
id	Incremental integer starting by zero	int
predicate	Wikidata Property id, e.g. 'P121'	string
predicate_label	Wikidata Property label (nullable)	string
predicate_desc	Wikidata Property description (nullable)	string

Table 6: `{variant}-{size}-{dataset_type}-predicates.csv` contains predicates. A predicate is a Wikidata property or a describing a connection. variant_index refers to the dataset id (detailed information is in our Github).

Dataset	Seed Nodes Categories
WikiLitArt	**Entire graph:** actor=150, composer=35, film=41, novelist=24, painter=59, poet=39, screenwriter=17, singer=72, writer=57
	Train: actor=105, composer=24, film=29, novelist=17, painter=42, poet=27, screenwriter=12, singer=50, writer=40
	Val: actor=23, composer=5, film=6, novelist=4, painter=9, poet=6, screenwriter=2, singer=11, writer=8
	Test: actor=22, composer=6, film=6, novelist=3, painter=8, poet=6, screenwriter=3, singer=11, writer=9
WikiCinema	**Entire graph:** actor=405, film=88
	Train: actor=284, film=61
	Val: actor=59, film=14
	Test: actor=62, film=13
WikiPro	**Entire graph:** actor=58, football=156, journalist=14, lawyer=16, painter=23, player=25, politician=125, singer=27, sport=21, writer=28
	Train: actor=41, football=109, journalist=10, lawyer=11, painter=16, player=17, politician=87, singer=19, sport=15, writer=20
	Val: actor=9, football=23, journalist=2, lawyer=3, painter=3, player=4, politician=19, singer=4, sport=3, writer=4
	Test: actor=8, football=24, journalist=2, lawyer=2, painter=4, player=4, politician=19, singer=4, sport=3, writer=4
WikiProFem	**Entire graph:** actor=141, athletic=25, football=24, journalist=16, painter=16, player=32, politician=81, singer=69, sport=18, writer=46
	Train: actor=98, athletic=18, football=17, journalist=9, painter=13, player=22, politician=57, singer=48, sport=14, writer=34
	Val: actor=21, athletic=4, football=3, journalist=4, painter=1, player=5, politician=13, singer=11, sport=1, writer=5
	Test: actor=22, athletic=3, football=4, journalist=3, painter=2, player=5, politician=11, singer=10, sport=3, writer=7

Table 9: Seed nodes categories for each dataset. "Entire graph" refers to using the seed nodes and generating the data without train-test-val splits. In train-test-val, each of the datasets is a single weakly connected graph.

A.2 Parameters for Running the WIKES Generator

Table 10 shows the parameters required for running the WIKES Generator. The table provides a description of the parameters and their default values, where applicable. A detailed explanation of how to run the generator can be found in our GitHub repository.

Field	Description	Datatype
subject	id key in Table 4	int
predicate	id key in Table 6	int
object	id key in Table 4	int

Table 7: `{variant}-{size}-{dataset_type}-triples.csv` contains triples. A triple is an edge between two entities with a predicate. variant_index refers to the dataset id (detailed information is in our Github).

Field	Description	Datatype
root_entity	entity in Table 4	int
subject	id key in Table 4	int
predicate	id key in Table 6	int
object	id key in Table 4	int

Table 8: `{variant}-{size}-{dataset_type}-ground-truths.csv` contains ground-truth triples. A ground-truth triple is an edge marked as a summary for a root entity.

Parameter	**Description**	**Default Value**
min_valid_summary_edges	Minimum number of valid summaries for a seed node	5
random_walk_depth_len	Depth length of random walks (number of nodes in each random walk)	3
bridges_number	Number of connecting path bridges between components	5
max_threads	Maximum number of threads	4
min_random_walk_number	Minimum number of random walks for each seed node, explained	100 for small, 150 for medium, and 300 for large
max_random_walk_number	Maximum number of random walks for each seed node	300 for small, 600 for medium, and 1800 for large

Table 10: Parameters for Running WIKES Generator

A.3 Technologies

Table 11 presents the versions of the technologies and configurations that we use in this work.

Table 11: Technology and Configuration Details for Daatset Generations

(a) Technologies Used: Software Versions and Data Sources

Technology	Version/Details
Java	Version 21
Spring Boot	Version 3
Docker	Version 24.0.8
Python	Version 3.10
PostgreSQL	Version 16.3
Neo4j	Version 5.20.0-community
Wikipedia XML Article Dump Files	Published by Wikimedia on 2023/05/01
Wikidata XML Article Dump Files	Published by Wikimedia on 2023/05/01

(b) Hardware- Spec: Specifications of the AWS EC2 Instance (r5a.4xlarge)

Specification	Details
vCPU	16 (AMD EPYC 7571, 16 MiB cache, 2.5 GHz)
Memory	128 GB (DDR4, 2667 MT/s)
Storage	500 GB (EBS, 2880 Max Bandwidth)

B Datasheet

B.1 Motivation

1. *For what purpose was the dataset created?* The motivation behind this dataset generator is to foster research on entity summarization and provide a tool to generate arbitrary-size datasets eschewing the cost of human annotation. We create an easy-to-use executable generator, in which users can create connected graphs by passing their desired seed nodes. Furthermore, we provide some curated datasets because there has not been a large dataset in this field, and consequently, scalable models have not been developed in this area. To address this issue, we generate large datasets alongside their medium and small versions for research. By studying current models on our generated datasets, we can pinpoint scalability issues of current models and nurture research in this area. Unlike previous benchmarks, we focus on generating datasets that maintain the graph structure of real-world graphs and on avoiding the introduction of bias. In Section 4, we study our dataset for different biases and compare it to the ESBM benchmark, which is the most recent entity summarization benchmark to date.
2. *Who created the dataset and on behalf of which entity?* The dataset is fully created by the authors of this paper.

B.2 Distribution

1. *Will the dataset be distributed to third parties outside of the entity (e.g., company, institution, organization) on behalf of which the dataset was created?* Yes, the dataset is open and fully accessible.
2. *How will the dataset be distributed (e.g., tarball on website, API, GitHub)?* The WIKES generator, WIKES benchmark and, the code for developing the baselines will be distributed through our GitHub repository.
3. *Have any third parties imposed IP-based or other restrictions on the data associated with the instances?* No.
4. *Do any export controls or other regulatory restrictions apply to the dataset or to individual instances?* No.

B.3 Maintenance

1. *Who will be supporting/hosting/maintaining the dataset?* The authors of the paper.
2. *How can the owner/curator/manager of the dataset be contacted (e.g., email address)?* The owner/curator/manager(s) of the dataset can be contacted through the following emails: Mohammad Sorkhpar (`msorkhpar@sycamores.indstate.edu`), Saeedeh Javadi (`saeedeh.javadi@studenti.polito.it`), and Atefeh Moradan (`atefeh.moradan@cs.au.dk`).
3. *Is there an erratum?* No, but the users can submit GitHub issues or contact the authors. If errors are found in the future, we will release errata on the main web page for the dataset (`https://github.com/msorkhpar/wiki-entity-summarization`).
4. *Will the dataset be updated (e.g., to correct labeling errors, add new instances, delete instances)?* Yes, the datasets will be updated whenever necessary to ensure accuracy, with announcements made accordingly. These updates will be posted on the main webpage for the dataset (`https://github.com/msorkhpar/wiki-entity-summarization`).
5. *If the dataset relates to people, are there applicable limits on the retention of the data associated with the instances (e.g., were the individuals in question told that their data would be retained for a fixed period of time and then deleted?)* N/A.
6. *Will older versions of the dataset continue to be supported/hosted/maintained?* Yes, older versions of the dataset will continue to be maintained and hosted.
7. *If others want to extend/augment/build on/contribute to the dataset, is there a mechanism for them to do so?* People are welcome to use what we provide for their own extensions.

B.4 Composition

1. *What do the instances that comprise the dataset represent (e.g., documents, photos, people, countries)?* Each instance is a document of a Wikidata item, containing some of its immediate neighbors sampled from the Wikidata Knowledge Graph.
2. *How many instances are there in total (of each type, if appropriate)?* We generate small, medium, and large versions of each of the four datasets, resulting in 12 datasets in total. The characteristics of these datasets are presented in Table 3 in section A. The number of entities ranges from 70k to 250k,

and the number of relations ranges from 120k to 470k, depending on the dataset size. See Section 3.4 for more information.

3. *Does the dataset contain all possible instances or is it a sample of instances from a larger set?* This is a sample instance of a larger dataset from Wikipedia and Wikidata. We generate small, medium, and large samples using Wikidata and Wikipedia. See section 3.2 for our sampling approach.
4. *Is there a label or target associated with each instance?* Yes, each target entity includes target variables.
5. *Is any information missing from individual instances?* No, we include all the information from the intersection of Wikipedia with Wikidata.
6. *Are there recommended data splits (e.g., training, development/validation, testing)?* We have generated train, validation, and test sets for each of the graphs, with 70 percent of the seed nodes in the train set, 15 percent in the test set, and 15 percent in the validation set. We recommend generating each split in a way that obtains a connected graph in each split. We propose the algorithm in section 3.4 to obtain such splits.
7. *Are there any errors, sources of noise, or redundancies in the dataset?* Not that we are aware of.
8. *Is the dataset self-contained, or does it link to or otherwise rely on external resources (e.g., websites, tweets, other datasets)?* The dataset is self-contained. Moreover, additional datasets can be generated using the WIKES generator that utilizes Wikipedia and Wikidata.
9. *Does the dataset contain data that might be considered confidential?* No.
10. *Does the dataset contain data that, if viewed directly, might be offensive, insulting, threatening, or might otherwise cause anxiety?* No.

B.5 Collection Process

1. *How was the data associated with each instance acquired?* Each instance in this dataset is acquired from raw XML data dumps published by the Wikimedia Foundation on 2023/05/01. The data is then cleaned and processed by our preprocessing module to extract entities, their relations, and their associated summaries. For more information on the summary annotation process, please refer to Section 3.1, "Summary Annotation."
2. *What mechanisms or procedures were used to collect the data (e.g., hardware apparatus or sensor, manual human curation, software program, software API)?* We utilized an AWS EC2 r5a.4xlarge machine to clean the data and build the datasets using a fully automated pipeline. However, Wikipedia admins and contributors can be considered indirect annotators of this dataset.
3. *Who was involved in the data collection process (e.g., students, crowdworkers, contractors) and how were they compensated (e.g., how much were crowdworkers paid)?* Regular employees and two master students were involved in the data collection process. However, the annotation process was fully automated, utilizing publicly available information.
4. *Does the dataset relate to people?* No.
5. *Did you collect the data from the individuals in question directly, or obtain it via third parties or other sources (e.g., websites)?* We sample the dataset from Wikimedia Foundation dump files on 2023/05/01.

B.6 Uses

1. *Has the dataset been used for any tasks already?* We have solely used the dataset for training and evaluating entity summarization baselines in the paper.
2. *What (other) tasks could the dataset be used for?* Even though the main target of this dataset is the entity summarization task, it might also be used for link prediction tasks in knowledge graphs.
3. *Is there anything about the composition of the dataset or the way it was collected and preprocessed/cleaned/labeled that might impact future uses?* This dataset is based on cleaned and annotated preprocessed information from Wikimedia Foundation published dump files on 2023/05/01. Using this preprocessed dump, one can generate a new dataset based on their desired seed node. However, to generate an updated version of the Wikidata knowledge graph, the preprocessing dumps should be regenerated from updated dump files. The instructions and code for this task are publicly available in our preprocessing URL.
4. *Are there tasks for which the dataset should not be used?* No.

C Experiments

We include the experiments in Section 4 for all of our datasets below.

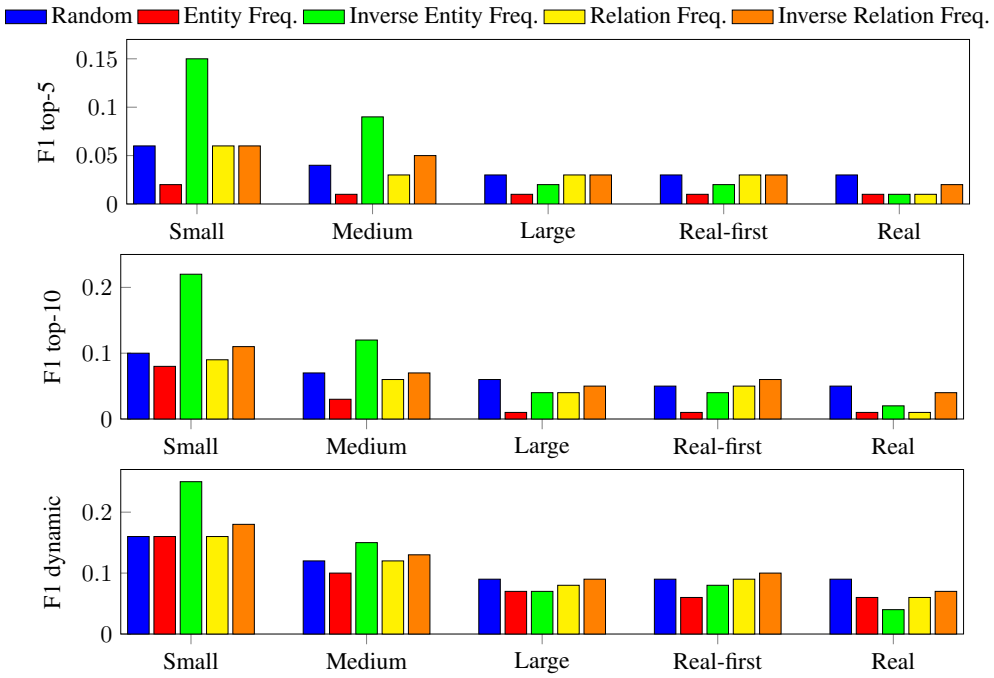

Figure 5: F1 for frequency statistics on WikiLitArt.

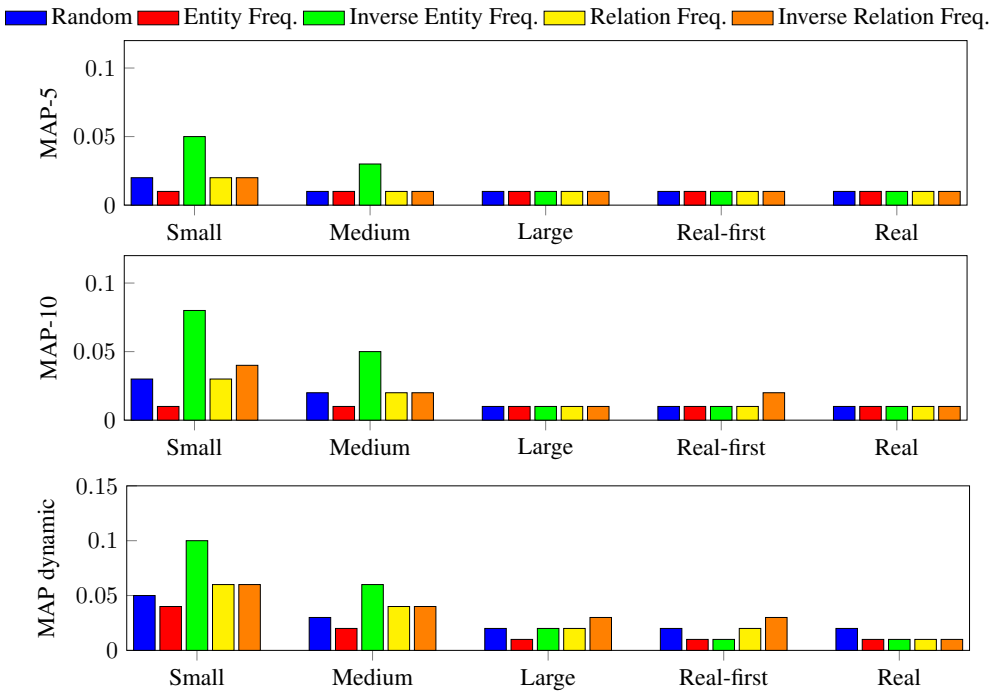

Figure 6: MAP for frequency statistics on WikiLitArt.

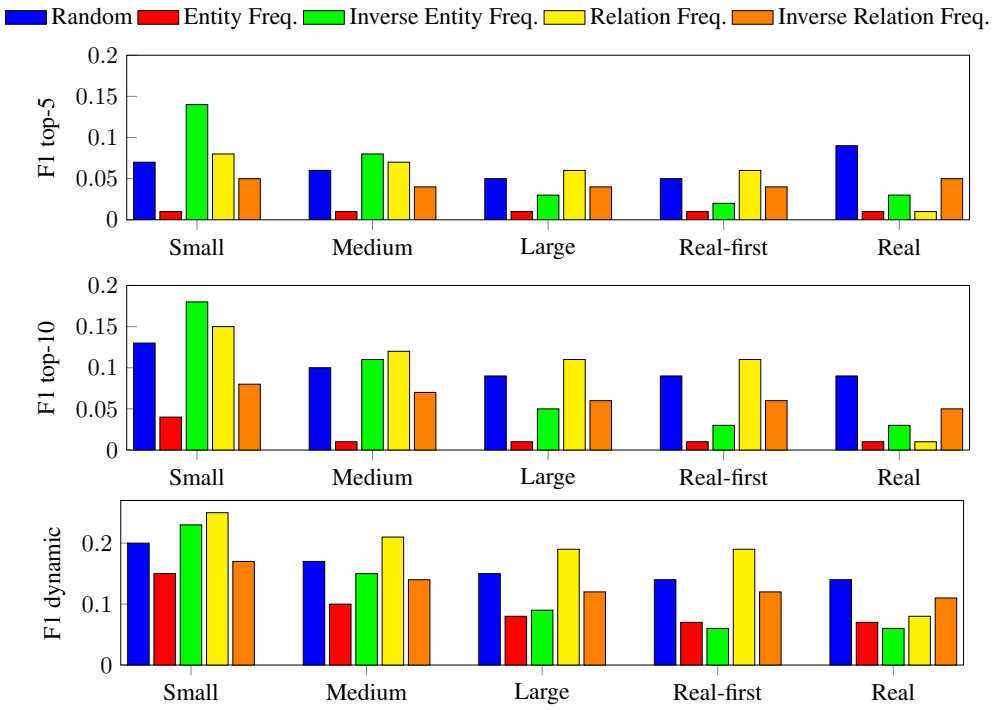

Figure 7: F1 for frequency statistics on WikiCinema.

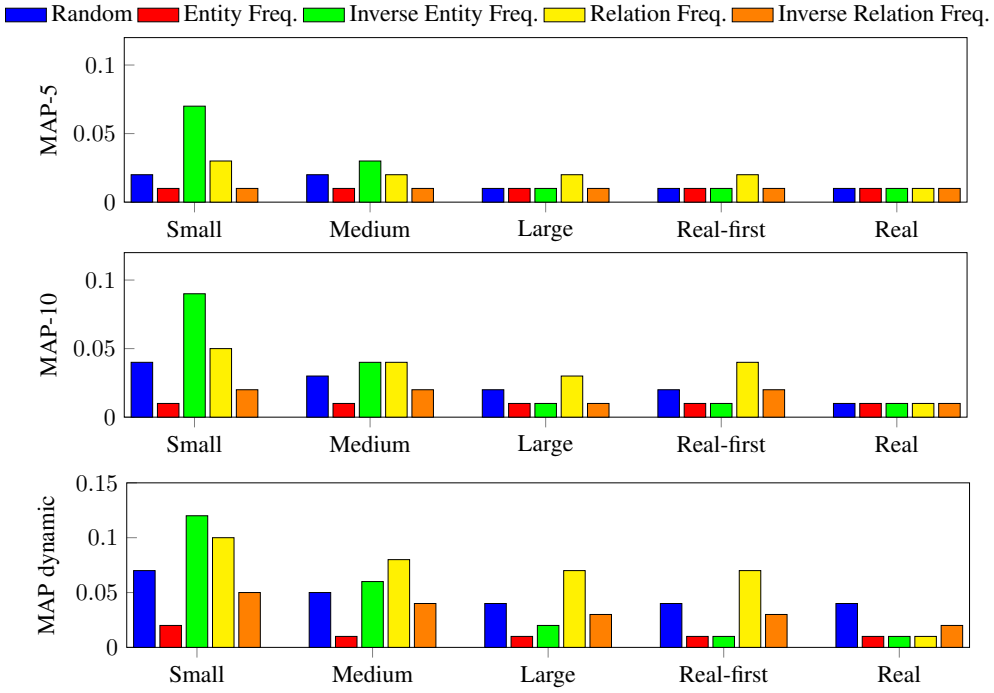

Figure 8: MAP for frequency statistics on WikiCinema.

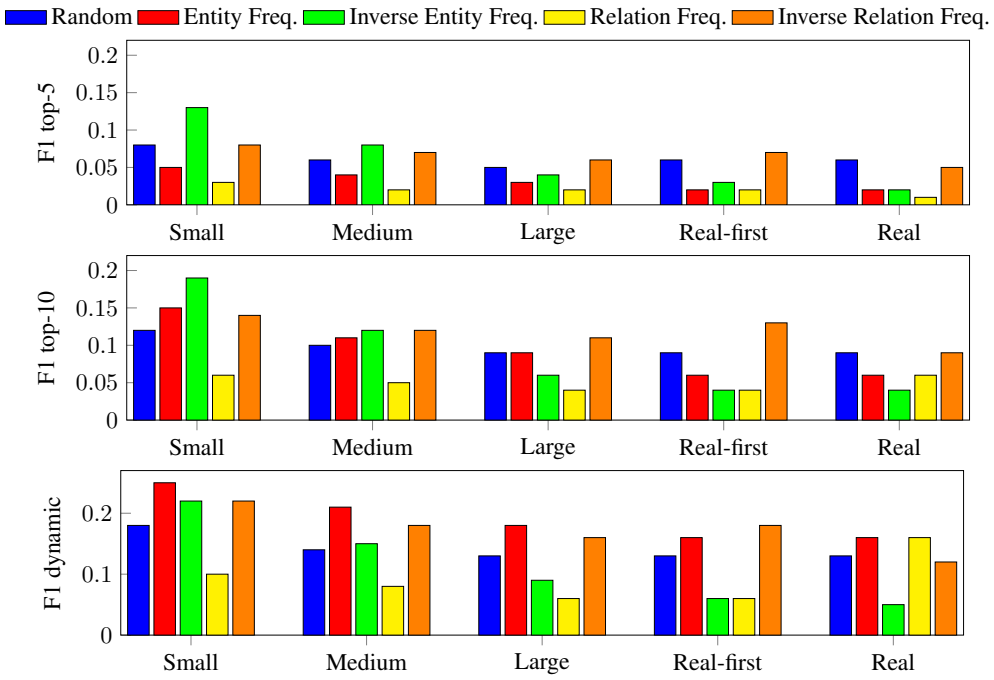

Figure 9: F1 for frequency statistics on WikiPro.

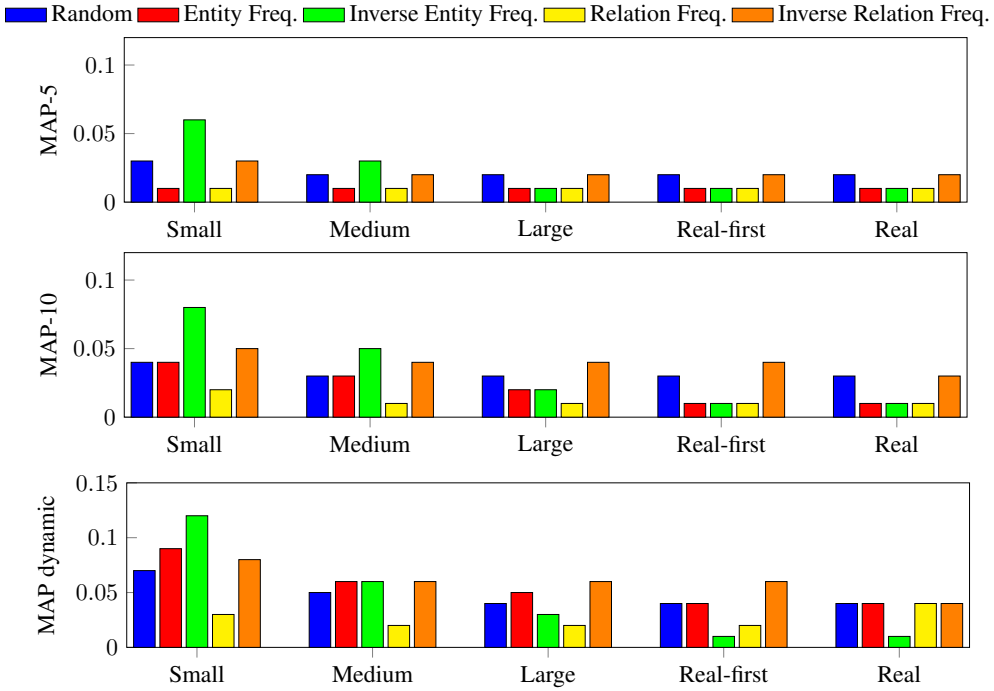

Figure 10: MAP for frequency statistics on WikiPro.

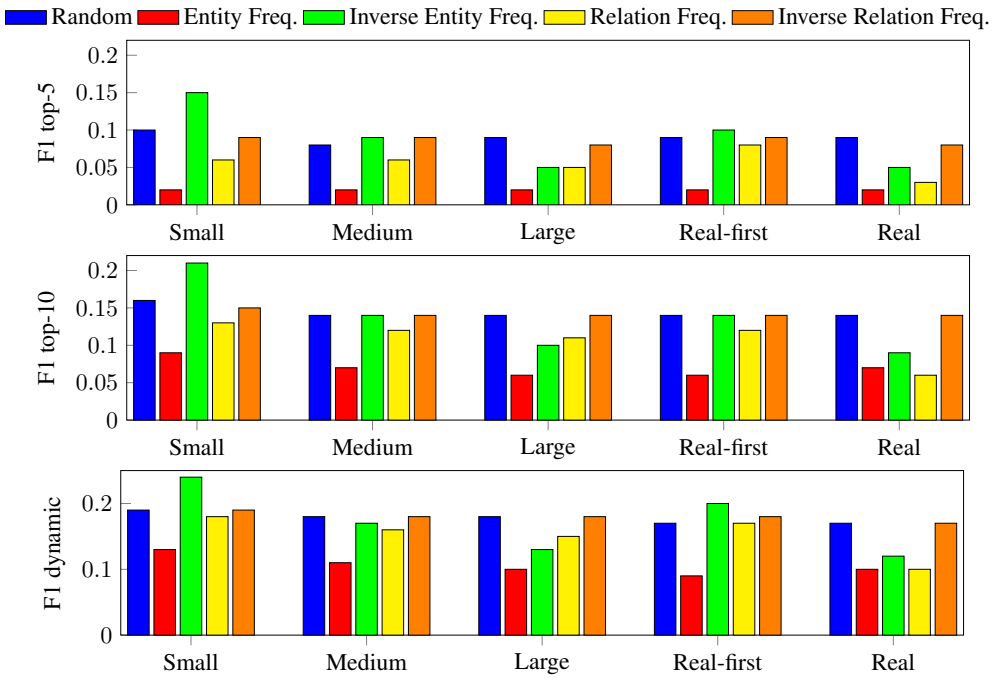

Figure 11: F1 for frequency statistics on WikiProFem.

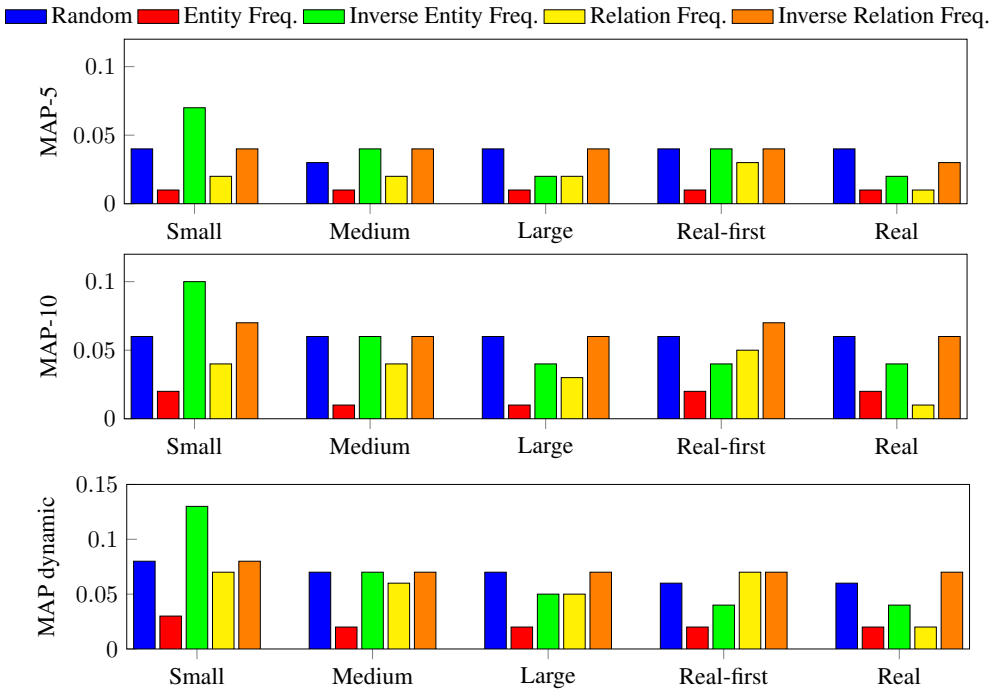

Figure 12: MAP for frequency statistics on WikiProFem.