NOSIBLE World - Data Dictionary V1.2

As of: 2026-07-13

This document describes every field in a NOSIBLE World event so that a technical reader can understand the data payload at a glance. Each line of events.ndjson is one world event. The fields are written in a fixed order so that a clipped row is still readable.

Input Field	Details
`event_id`	**Definition:** A 72-character identifier that uniquely names the event. It bakes in the date, language, country, schema version, the full ontology block, and a short random suffix. **Type:** string **Shape:** `{date}_{language}_{country_iso}_v{schema_version}_{ontology_code}_{random_code}`
`version`	**Definition:** The schema version of this row. We bump it whenever the structure changes in a way that breaks older readers. **Type:** string **Value:** "1.2"
`has_tickers`	**Definition:** True when the event mentions one or more companies we matched to a ticker. Used by the frontend and API as a quick filter. **Type:** Boolean

`event`	**Definition:** The headline view of the event: a short title and one-paragraph description written in the event's own language, plus its date and place. **Type:** object **Fields:** • `date` (str) - the UTC date of the event, YYYY-MM-DD • `title` (str) - up to 80 characters, in the event's language • `description` (str) - up to 420 characters, in the event's language • `continent` (str) - one of six continents • `region` (str) - one of thirty world regions • `country` (str) - the full English name of the country • `language` (str) - ISO 639-1 two-letter code
`coordinate`	**Definition:** Where the event happened on a map. **Type:** object or null **Fields:** • `lat` (float) - latitude in WGS84, 4-decimal precision • `lng` (float) - longitude in WGS84, 4-decimal precision • `source` (str) - how the lat + lon was inferred
`signals`	**Definition:** Three quick-read labels the frontend leads with. **Type:** object **Fields:** • `materiality` (str) - high, medium, or low. • `sentiment` (str/null) - positive, neutral, or negative • `forward_looking` (str) - forward looking or not-forward looking • `time_horizon` (str/null) - intraday, days, weeks, months, quarters, or years • `materiality_score` (float) - raw five-axis score that drives the materiality bucket

`coverage`	**Definition:** How widely the event was reported. **Type:** object **Fields:** • `total_coverage` (int) - total articles about this event • `total_netlocs` (int) - number of distinct news outlets • `coverage_concentration_index` (float) - how concentrated coverage is across outlets; lower means more distributed

`entities`	**Definition:** The named things mentioned in the event, with how often each appears.
	Type: object
	Shape:
	`{`
	`  TYPE: {`
	`  text: count`
	`  }`
	`}`
	Note: Types are `PERSON`, `ORG`, `GPE`, `LOC`, `PRODUCT`, `EVENT`, `DATE`, and `MONEY`. Entries are sorted by descending count within each type.

`tickers`	**Definition:** Companies named in the event, matched to their stock and identifier records. One row per company, ordered by descending `match_count`. **Type:** array of objects **Fields:** • `name` (str) - the company's canonical name • `website` (str) - primary domain • `wiki_qid` / `wiki_page` (str) - Wikidata identifiers • `linkedin` (str) - LinkedIn company URL • `name_eodhd` (str) - display names from EODHD • `name_figi` (str) - display names from OpenFIGI • `name_kgid` (str) - display names from Google • `name_lei` (str) - display names from LEI • `name_orig` (str) - display names from NOSIBLE • `name_wiki` (str) - display names from Wikipedia • `isin` / `lei` / `figi` (str) - standard security identifiers • `ticker_wiki` (str) - ticker from Wikipedia • `ticker_figi` (str) - ticker from FIGI

	• `ticker_eodhd` (str) - ticker from EODHD • `ticker_serp` (str) - ticker from search engine result page • `match_string` (list of str) - the ORG string(s) from the event that were used to match this public company and ticker. • `match_count` (int) - summed entity mention count for the matched strings across all the considered documents.

`ontology`	**Definition:** Thirteen taxonomies that classify the event from many angles. Each slot is a list of the top three most likely classifications recorded as `{label, cosine}` pairs, where the cosine is how confident the classifier was. Every slot also carries a below_floor Boolean that flags scores under the assignment threshold.
	
	Type: object
	Fields:
	
	• `iptc_genre` (required) - List of top 3 IPTC genre labels
	• `media_frame` (required) - List of top 3 media frames
	• `iptc_media_topics` (required) - List of top 3 media topics.
	  ○ level_1: the implied level 1 classification.
	  ○ level_2: the implied level 2 classification.
	  ○ level_3: the predicted level 3 classification.
	• `iab` (required) - List of top 3 IAB category labels.
	  ○ tier_1: the implied tier 1 classification.
	  ○ tier_2: the implied tier 2 classification.
	  ○ tier_3: the implied tier 3 classification.
	  ○ tier_4: the predicted tier 4 classification.
	• `ekman7_emotion` (nullable) - List of top 3 Ekman emotions.
	• `schema_org_event` (nullable) - List of top 3 Schema.org events.
	• `gics` (nullable) - List of the top 3 most likely GICS classification.
	  ○ sector: the implied GICS sector.
	  ○ industry_group: the implied industry group.
	  ○ industry: the implied industry.
	  ○ sub_industry: the predicted sub-industry.
	• `icd11` (nullable) - List of top 3 ICD-11 classifications.
	  ○ chapter: the implied ICD-11 chapter.
	  ○ block: the implied ICD-11 block.
	• `sportsml` (nullable) - List of top 3 most likely SportsML classifications.
	  ○ sport: the implied sport.
	  ○ event_type: the predicted event type.
	• `nosible` (nullable) - List of top 3 NOSIBLE classifications.

	○ category: the implied top-level category. ○ subcategory: the implied subcategory. ○ event: the predicted event. ● `emdat` (nullable) - List of top 3 EM-DAT classifications. ○ disaster_group: the broad disaster grouping. ○ disaster_subgroup: the implied disaster subgroup. ○ disaster_type: the predicted disaster type. ● `plover` (nullable) - List of top 3 PLOVER classifications. ○ event: the predicted event type. ○ mode: the primary mode or mechanism associated. ○ context: the contextual classification of the event. ● `mitre_attack` (nullable) - List of top 3 MITRE ATT&CK classifications. ○ tactic: the highest-level adversary objective. ○ technique: the implied attack technique ○ subtechnique: the predicted attack sub-technique. ● `asset_class` (nullable) - List of top 3 Asset Class classifications. ○ main_class: the highest-level asset class. ○ sub_class: the predicted asset subclass. ● `sdgs` (nullable) - List of top 3 UN Sustainable Development classifications. ○ goal: the implied Sustainable Development Goal. ○ target: the predicted SDG target within the goal.

`provenance`	**Definition:** Representative source articles for the event.
	Type: array of objects
	Shape:
	`[`
	`{`
	`event_score,`
	`url,`
	`title,`
	`description,`
	`snippets`
	`},`
	`...`
	`]`
	
	Fields:
	
	● `event_score` (float) - relevance score of the article to the event, between 0 and 1

	• `url` (str) - link to the source article • `title` (str) - the article's headline • `description` (str) - short summary of the article • `snippets` (list of str) - body-text excerpts pulled from the article **Note:** Ordered by descending event_score.

`similar`	**Definition:** The closest related events from the previous seven days, if any were found above the similarity threshold.
	Type: object
	Shape:
	`{`
	`  event_id: score,`
	`  ...`
	`}`

`oai_vector`	**Definition:** The OpenAI `text-embedding-3-large` vector of the title and description, stored as a comma-separated string. Used for semantic search. **Type:** string **Dimensions:** 3072

`timestamps`	**Definition:** An object containing all published timestamps we observed from webpages associated with this specific event. It includes the earliest seen timestamp (first_seen), the latest seen timestamp (last_seen), the most seen timestamp, and a histogram showing the distribution of timestamps.
	Type: object
	Shape:
	`{`
	`  “first_seen”: str,`
	`  “last_seen”: str`
	`  “most_seen”: str`
	`  “hourly_counts_utc”: {`
	`  TIME: count,`
	`  ...`
	`  TIME: count`
	`  }`
	`  “utc_all_seen”: [`
	`  TIME,`
	`  ...`
	`  ]`
	`  “raw_all_seen”: [`

	`TIME,`
	`...`
	`]`
	`}`

`extra`	**Definition:** Extra signal probabilities that don't belong in the picked-label `signals` block. **Type:** object **Fields:** • `extra.signals.sentiment` (dict/null) - three probabilities (`prob_positive`, `prob_negative`, `prob_neutral`) that sum to one, or null on classifier failure • `extra.signals.forward_looking` (dict/null) - two probabilities (`prob_forward`, `prob_not_forward`) that sum to one, or null on classifier failure