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.

Categorization of scientific publications using large language models as applied to atomic spectroscopy

Manas Pandey
Indian Institute of Technology Kanpur
Alexander Kramida
University of Maryland, College Park
Yuri Ralchenko
University of Maryland, College Park
Dmitry Budker
budker@uni-mainz.de

Johannes Gutenberg University Mainz

Article

Keywords:

Posted Date: October 26th, 2026

DOI: https://doi.org/10.21203/rs.3.rs-11468277/v1

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

Additional Declarations: No competing interests reported.

Categorization of scientific publications using large language models as applied to atomic spectroscopy

Manas Pandey [1], Alexander Kramida [2,3,4], Dmitry Budker [5,6,7,8,*], and Yuri Ralchenko [2,3,4]

[1]Indian Institute of Technology, Kanpur, Uttar Pradesh 208016, India
[2]Department of Astronomy, University of Maryland, College Park, MD 20742, USA
[3]High Energy Astrophysics Laboratory, NASA/GSFC, Greenbelt, MD 20771, USA
[4]Center for Research and Exploration in Space Science and Technology, NASA/GSFC, Greenbelt, MD 20771, USA
[5]Johannes Gutenberg-Universität Mainz, Mainz 55122, Germany
[6]Helmholtz Institute Mainz, Mainz 55099, Germany
[7]GSI Helmholtzzentrum für Schwerionenforschung GmbH, Darmstadt 64291, Germany
[8]Department of Physics, University of California, Berkeley, CA 94720-7300, USA
[*]Corresponding author: budker@uni-mainz.de

October 22, 2026

Abstract

We present an automated framework employing large language models to categorize atomic spectroscopy literature for NIST bibliographic databases. The system processes native PDFs and supplementary materials via instruction-driven reasoning agents to assign structured search keywords encompassing target species, subject codes, and experimental methodologies. Validation against human-curated datasets demonstrates high precision in elemental species identification, while establishing that granular subject code assignment remains the primary source of categorization error.

Dedication: Dedicated to the memory of William C. Martin, an atomic spectroscopy pioneer

1 Introduction

The ongoing revolution in artificial intelligence (AI) is changing how research is done, and it reaches every stage of the process. In the natural sciences, which rest on an experimental foundation, the value of that research depends on the availability of reliable and carefully curated data. Atomic spectroscopy is a quintessential example.

For many years, the atomic spectroscopy program at the National Institute of Standards and Technology (NIST) has focused on producing critically evaluated sets of atomic data — energy levels, wavelengths, transition probabilities and line broadening parameters — that serve as benchmarks for extremely diverse fields of science. The NIST online databases, and above all the flagship Atomic Spectra Database (ASD) [1], have justifiably become the most trusted sources of precise atomic parameters for astrophysics, fusion science, extreme ultraviolet lithography, and other research areas.

Alongside the numerical databases, NIST maintains bibliographic databases that record where the data came from (see `https://physics.nist.gov/asbib` and, for energy levels, `https://physics.nist.gov/elevbib`). These are indispensable tools for the analysis of atomic spectra, but only if the articles in them are well categorized. Categorization here means the assignment of carefully designed search keywords, so that a user can retrieve the literature containing original data on an exact topic of interest: a chemical element, a degree of ionization, a specific subject and method of treatment, or a subject of general interest such as plasma diagnostics, exotic atoms, or the variation of fundamental constants.

Keeping such a database current is laborious. About 500 new papers on atomic spectroscopy are

published each year, and they have to be found among the much larger number of articles published across all of physics. Journal subject codes and publisher keywords narrow the search, but the final judgment is manual. For every candidate the curator must obtain the full text and the supplementary materials, read all of it, decide whether it falls within the scope of the database, assign the proper search keywords, and enter the material into the database.

These tasks were carried out by hand for the past century by renowned atomic spectroscopists such as Charlotte E. Moore, William C. Martin, Wolfgang L. Wiese and Jeffrey R. Fuhr, and the results were published on paper (see, for example, [4, 2]). The NIST Atomic Spectroscopy Group continued the work in the same manner after the printed bibliographies had been transferred to the online bibliographic databases incorporated in the NIST ASD [1].

The work requires considerable skill, and much of it consists of solving small puzzles. It is necessary, for example, to establish whether the data in an article are really new or are merely quoted from other sources. The authors' own statements about the scope of their work cannot always be trusted in our experience, abstracts often reflect the original intent of a study rather than the content of the finished article. At present, only very few people possess the expertise required for maintaining and expanding such bibliographic databases.

In this work we develop a tool that categorizes published articles on atomic spectroscopy using large language models, with the current expert knowledge written into the instruction given to the model. For newly located and downloaded papers it produces the categorization needed for insertion into a bibliographic database. It removes the most laborious part of the process, since it reads the whole of each paper together with its supplementary material. A separate and related effort, to be described elsewhere, is directed at extracting the quantitative data themselves rather than at categorizing the papers that contain them.

We find that large language models presently available can analyze the text, figures and tables of a published article and assign relevant search keywords to it. A set of Python codes invoking artificial intelligence agents has been written for this purpose. The main code holds the prompt, which carries a detailed description of the topical structure of the atomic spectroscopy discipline together with the instructions for assigning the corresponding search keywords. It accepts one or more sets of PDF files with their supplementary materials, with no intrinsic limit on the number of papers, and returns for each paper a text file containing the assigned keywords.

The tool supplements human work rather than replacing it, and Sec. 3.6 states plainly what our measurements do and do not establish. Its clearest advantage is on volume and on completeness of reading: it does not tire, it does not skip the supplementary tables, and it therefore sometimes recovers data that a human expert had missed. The advantage is largest where the sheer amount of material makes manual assessment impractical. Assessing the large solar spectral atlas [3], 811 pages long, took days of expert time by hand; with the present tool it was done in a few minutes.

2 Methodology

This section describes the pipeline that turns a published article into a set of search keywords. Figure 1 summarizes the flow of information. In outline: the article and its supplementary materials are collected and, where necessary, converted into a form the model can read; they are passed in a single request to a large language model that has been given a detailed instruction; and the keywords the model returns are written to a text file in the format used by the bibliographic database.

2.1 What the model is given

The article itself is passed to the model natively as a PDF, and not as extracted plain text. Figures, plotted spectra and the internal structure of tables therefore remain visible to the model. This matters because much of the numerical content that decides a keyword is carried in exactly those places.

Supplementary materials are far more heterogeneous, and they are converted to text before being sent. A recursive pre-processor walks the supplementary directory and converts each file according to its type: PDF, Word and Excel documents through the corresponding libraries; ZIP and compressed tar archives by expanding them and treating each member in the same way; and everything else by decoding it as text, which covers data tables, source code and auxiliary documentation. Files that cannot be decoded are skipped, and the skip is recorded in place so that the model is not misled into assuming the material was empty.

Nothing is summarized, and the article is never abridged. The complete document and the full supplementary text are presented in a single request, since a keyword can be assigned correctly only after the whole of the paper has been understood in detail. There is one guard against pathological cases: the

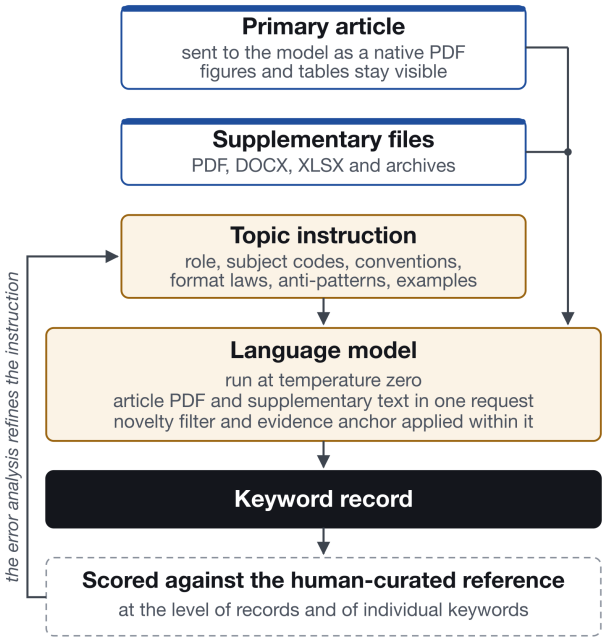


Figure 1: Flow of information through the categorization pipeline. The primary article reaches the model natively as a PDF, while the supplementary files are reduced to text by a recursive pre-processor; both arrive in the same request as the topic instruction. The novelty filter and the evidence anchor of Sec. 2.3 are applied within that single pass, not as a separate stage. The three topic instructions are applied to the same paper in turn, and a deterministic post-processor normalizes notation without ever deleting a keyword. Amber marks the language model and its instruction, the dashed box the human-curated benchmark, and the dark box the record that the system produces.

extracted supplementary text is truncated at three million characters. In the corpus used for validation, one paper's supplementary archive exceeds this limit, extracting to about ten million characters.

2.2 The instruction

The domain knowledge of the system is held in its instruction rather than in trained weights. For the Energy Levels topic the instruction is a document of some 18,000 characters. It fixes the role that the model is to adopt, that of an expert atomic physicist and bibliographer [5]. It then lists the subject codes with their definitions, their permitted method types and their boundary cases; states the notation conventions for naming spectra; and lays down the format rules that keep the output machine-readable. A catalogue of anti-patterns closes the document, recording mistakes seen in earlier runs with the correct treatment set beside each one.

Holding this knowledge in a prompt rather than in weights has a practical consequence: the domain experts can read it, version it and correct it themselves, and no retraining is involved.

Generation is run at temperature zero, which keeps the behaviour of the model stable across large numbers of disparate papers. It does not make the system strictly deterministic, since the model allocates a variable internal reasoning budget; run-to-run variation has not been measured, as noted in Sec. 3.6.

2.3 The form of a keyword

A keyword occupies one line and has three fields separated by colons, for example `Fe II: EL: E`. The first field names the species: a single spectrum such as `Fe II`, a compact range such as `K-Fe I`, or an isoelectronic sequence such as `Ar-U Be-like`. The second is the subject code, which identifies the class of data reported. The third is the method type, recording whether the values are experimental (`E`), theoretical (`T`), or obtained by other means (`O`). Codes are never combined on one line, so a paper reporting both measured wavelengths and classified lines for the same species yields two lines rather than one.

Two rules govern whether a code is assigned at all, and both exist to suppress false positives. The *novelty filter* admits only data that the paper presents as its own new result. Values quoted from earlier literature, values reproduced in a comparison column, and values taken from a database for context are all rejected. Its operational form is a question put to every candidate keyword: would a future researcher cite *this* paper in order to retrieve *this* value? The second rule anchors each code to evidence. Before a code is emitted, the model must be able to point to the particular table or section in which the corresponding data are actually reported, which excludes topics that a paper merely touches upon without producing numbers of its own. Section 3.4 shows that over-assignment nevertheless remains the largest single source of error, so these rules mitigate the problem rather than solve it.

2.4 The three topics

The bibliographic database divides atomic spectroscopy into three parts: Energy Levels (EL), Transition Probabilities (TP) and Line Broadening (LB). Each part has its own subject codes — fifteen for EL, eleven for TP and seven for LB — and its own conventions, and therefore its own instruction file. A

further small set of general-interest codes marks subjects that cut across all three, such as atomic clocks, exotic atoms, and searches for physics beyond the standard model.

A paper that needs EL keywords commonly needs TP or LB keywords as well. The pipeline therefore applies the three instructions in turn to the same paper and collects the results into a single record. All three parts are validated quantitatively against a human-curated reference set, as described in Sec. 3.

3 Validation

We validated the pipeline against reference sets of papers that had been categorized by hand by an experienced curator of the NIST bibliographic databases, one set for each of the three topics. Below we describe the reference sets, the way the agreement with the reference is measured, the results, and the conclusions we draw from them. The three topics are validated in the same way, each against a single configuration of its instruction; no ablation across prompt variants is reported.

3.1 Terminology

Three terms are used throughout, and each of them denotes a different object.

- A *record* is the complete set of keywords belonging to one paper. Each paper has one reference record, written by the curator, and one generated record, produced by the system.
- A *keyword line* is one line of a record, for example `Na-S I: CL: E`. It names a species or a range of species (here the neutral atoms from sodium to sulfur), one subject code (here `CL`, classified lines), and one method type (here `E`, experimental).
- A *unit* is a single element–ion combined with one subject code and one method type, for example (`Na I`, `CL`, `E`). A keyword line expands into as many units as it names species, so the line above yields six units.

Records and keyword lines are the objects that the curator writes and the system emits. Units are an auxiliary construct used only for scoring: they allow a keyword line that is partly right to receive partial credit instead of being counted as a plain failure.

3.2 The reference sets

Hundred papers were processed for TP and LB and 99 for EL, curated by the same experienced curator of the NIST bibliographic databases. On EL, 91 carry at least one reference keyword line, 237 keywords in total, and 8 were examined by the curator and found to contain no original data within the scope of the database, so their reference records are empty. On TP, one paper carries no reference entry and is excluded on the same grounds, leaving 99 evaluable, where 96 carry at least one reference line, 133 in total, and 3 are empty. All 100 LB papers are evaluable and carry at least one reference keyword line, under the counting convention given in Sec. 3.4.

The empty records are kept in the set deliberately. Declining to assign anything is part of the task, and a system that always emits something would create more work than it saves. A paper of this kind counts as handled correctly when the system also emits nothing.

Generation and scoring are kept strictly separate. All numbers below are computed after the fact by deterministic code that makes no model calls, so they can be recomputed from the stored records alone. Before a generated line is compared with a reference line, both are passed through a deterministic normalization of notation, so that equivalent spellings of the same species set, such as `Te IV,V` and `Te IV-V`, are treated as equal. Only genuine differences of content are therefore counted as errors.

3.3 How agreement is measured

Agreement is measured at two levels, because two different questions are being asked.

The first level is the record. Here we count three things, in decreasing order of strictness. A record is *exact* when every reference line was matched and nothing else was emitted; this is the condition under which the output could enter the bibliographic database without any review. A record is *complete* when every reference line was matched but extra lines may also be present; this measures coverage alone. A record is *missed* when not a single reference line was matched. The last case matters most in practice: an imperfect but non-empty record still allows the paper to be found, whereas a missed paper is invisible to a user searching the database.

The second level is the unit. Each record is expanded into units as defined in Sec. 3.1, with three conventions: element ranges are expanded, so that `Cr-Fe I` becomes three units; isoelectronic sequences such as `Ar-U Be-like` are kept whole, because they denote a physical relationship rather than a list; and an isotope is matched to its base ion. For the resulting units we report the precision P, the fraction of the emitted units that are correct, the recall R, the fraction of the reference units that were recovered, and their harmonic mean F_1. These are computed under

three nested match criteria: the element–ion alone, then the element–ion together with the subject code, then all three components. Because the criteria are nested, the drop from one to the next isolates the decision that fails.

We quote both a macro average, in which each paper contributes equally, and a micro average, in which all units are pooled so that papers with many species dominate. The macro average is our primary figure, since the practical unit of work is one paper.

3.4 Results across the three topics

Two kinds of error dominate, and both appear on all three topics. The first, and by far the more frequent, is assigning a subject code that implies the presence of valuable data to a paper that only quotes those data from the literature or from a database, whereas the bibliographic system is meant to point to original sources. The second is more mechanical: the boundary of an element range is misplaced, so that one element is silently included or omitted.

For Line Broadening, the general-interest code GENINT is counted below like any other keyword, some LB reference records consist of a GENINT line and nothing else and would otherwise be misclassified as empty. It is, worth noting, a somewhat more subjective code than the topic-specific ones, as it records how broadly a paper's relevance extends rather than a definite value for a definite species.

Table 1 places the three topics side by side at the level of whole records. Table 2 gives the corresponding unit-level precision, recall and F_1.

Table 2 localizes the first error type. Identifying *which* species a paper reports data for is close to solved on every topic, at a macro F_1 of 81.7% for EL, 83.2% for TP and 74.2% for LB. Adding the subject code to the match criterion produces the largest single drop in macro F_1 at every topic — 13.2 points for EL, 13.1 for TP, and 14.3 for LB — so selecting the correct code is the dominant source of error regardless of topic or code list. Precision lies below recall at every level and every topic, the quantitative expression of the first error type: the system assigns codes the curator did not assign more often than it misses codes the curator did assign.

Method type is close to solved on top of that for EL and TP, costing a further 0.4 and 1.1 macro points respectively, but not for LB, where it costs a further 6.1 macro points and, more strikingly, drops the micro F_1 from 65.4% to 44.7%. Inspection of the individual mismatches shows why: 70 of them carry a reference method type of `CM` or `CP`, comment and compilation papers, a value the LB instruction never lists as available, so the model cannot produce it regardless of its judgment. This is a gap in the topic's instruction rather than a limitation the model could close on its own, and it falls on a handful of wide-species compilation papers, which is why it shows far more strongly in the micro average, which those papers dominate, than in the macro one.

The micro averages lie below the macro averages for EL and TP, which says the system performs worse on papers carrying many species than on ordinary ones; this is consistent with the second error type, since a wide element range offers many more places at which a boundary can be misplaced or over-extended. On the complementary task the system does well on EL and TP: it correctly left empty 7 of 8 evaluable EL records and 2 of 3 TP records that should be empty, so it is capable of declining, which is the prerequisite for the novelty filter of Sec. 2 working at all. No LB record has an empty reference under the counting convention above, so this check does not apply to that topic.

3.5 An approach that was abandoned

One approach was tried and then discarded. We mention it because it shows where the difficulty of the task lies.

It was an iterated verification, in which the model re-reads the article together with its own candidate keywords and is instructed to remove any that the paper does not support as an original result. Since precision was the main weakness, this looked like the obvious remedy. The verifier, however, over-pruned: it deleted correct keywords along with incorrect ones, and the number of exact records fell from 16 to 14 in the earlier round in which it was tested, scored there over 91 papers. The failure appears to be structural rather than a matter of wording, since the model has no evidence on the second pass that it lacked on the first; applying the procedure repeatedly would be expected to compound the loss rather than to converge. It was replaced by a deterministic post-processor that normalizes notation only and by construction can never delete a keyword. This guarantees the mechanical part of the formatting, although its measured effect on the benchmark was nil.

3.6 What the validation does and does not establish

Four limitations should be stated plainly.

All three topics are validated against a single configuration of their instruction; no ablation across prompt variants was run, so how sensitive the re-

Table 1: Agreement with the human-curated reference at the level of whole records, one configuration per topic. Each cell is given as (count)/(denominator); denominators differ by topic and are stated in each cell rather than in the header. *Extra* is a count of keyword lines, not a fraction.

	Records				Keyword lines	
Topic	Exact	Correct empty	Complete	Missed	Matched	Extra
EL	19/91	7/8	41/91	24/91	146/237	90
TP	38/96	2/3	53/96	39/96	79/133	91
LB	8/100	—	53/100	30/100	88/169	298

Table 2: Precision, recall and F_1 for individual keyword units, one configuration per topic (EL: 99 papers; TP: 99 papers; LB: 100 papers). Each row adds one component to the match criterion, so the drop between rows isolates the failing decision. The macro average weights each paper equally; the micro average pools all units and is therefore dominated by papers naming many species. A record that should be empty scores unity when the system also emits nothing, and zero otherwise.

		Macro-averaged			Micro-averaged		
Topic	Match criterion	P	R	F_1	P	R	F_1
EL	Element–ion only	82.8%	82.9%	**81.7%**	85.6%	55.6%	67.4%
	+ subject code	68.3%	75.3%	**68.5%**	65.8%	57.8%	61.6%
	+ method type	67.8%	74.6%	**68.1%**	65.3%	57.6%	61.2%
TP	Element–ion only	82.7%	84.3%	**83.2%**	92.9%	59.3%	72.4%
	+ subject code	67.7%	76.4%	**70.1%**	71.2%	60.1%	65.2%
	+ method type	66.7%	75.1%	**69.0%**	70.5%	59.5%	64.5%
LB	Element–ion only	66.4%	90.8%	**74.2%**	75.2%	78.3%	76.7%
	+ subject code	52.5%	80.7%	**59.9%**	62.5%	68.6%	65.4%
	+ method type	46.3%	74.4%	**53.8%**	41.4%	48.5%	44.7%

sults are to changes in the instruction remains open. The central diagnosis, subject-code assignment as the dominant source of error, reproduces on all three topics (Sec. 3.4), which is evidence the finding is not an artifact of one topic's code list, but it does not by itself establish how much of the remaining error a revised instruction could remove. Part of LB's method-type deficit is itself a known instruction gap rather than a model limitation: the reference set uses two method types, `CM` and `CP`, that the LB instruction does not permit the model to emit (Sec. 3.4); this has not yet been corrected and re-measured.

The novelty filter is not measured in isolation, since the benchmark scores complete records. Its effect is visible only indirectly, through the result on the deliberately empty records.

Consistency has not been characterized on either side of the comparison. Generation across the three topics was not necessarily performed with the same model or model version, and neither the agreement between different LLMs nor the run-to-run variation of one model on the same input has been measured. On the human side, the reference records were prepared by a single curator, and no second independent reading exists against which the variability of the reference itself could be estimated.

Finally, the system does not replace the curator's judgment. Where it demonstrably outperforms manual work is in volume and in completeness of reading: it does not tire, it does not skip the supplementary tables, and it can hold a very large document in view at once. In this respect it sometimes recovers data that a human expert had missed. Deciding whether a given result is genuinely new, however, remains a matter of expert judgment, and it is exactly the decision on which Table 2 shows the system to be weakest, consistently across all three topics.

4 Discussion and Outlook

We have developed and characterized an LLM-based system that assigns bibliographic search keywords to atomic spectroscopy publications. The presently developed system incorporates the knowledge of human experts on the scope, structure, and semantics of

the atomic spectroscopy discipline, condensed as a prompt for an LLM rather than into trained weights, and it converts a PDF together with its supplementary materials into a structured keyword record following NIST conventions. It makes tractable large documents that were previously assessed only at great cost in expert time.

A remark on generality is in order, since it would be easy to overstate. The architectural principles used here like native ingestion of the document, a domain taxonomy held outside the code, a novelty filter, and a scoring scheme that localizes rather than merely quantifies the failure are not specific to atomic spectroscopy, and we expect them to transfer. The system itself does not. It is tied closely to the semantics and conceptual structure of the discipline and to the conventions and limitations of the targeted bibliographic system, and essentially all of its domain content lives in prompts written by specialists in that discipline. A comparable system for another field would reuse the architecture and rewrite the content. In that limited but real sense the present work offers a prototype for the future of LLM-assisted scientific bibliographic curation.

Future work can extend these results in a number of directions. The most direct is to attack the identified weakness rather than the aggregate: since the loss is concentrated almost entirely in the assignment of subject codes (Table 2), effort is better spent on the boundaries between codes than on general refinement of the prompt. A library of worked examples retrieved per paper, rather than fixed in the prompt, could help with the cases where the system currently fails, such as rare transition spectra, ambiguous experimental notations, and the code boundaries themselves. Distributing work across parallel agents, each responsible for a subset of the keywords, would improve throughput, although the verification experiment of Sec. 3 counsels caution about architectures in which one model pass judges another.

While the current methodology relies on pretrained large language models, recent advances in parameter-efficient small language models present a viable alternative trajectory. Fine-tuning a compact model exclusively on a rigorously verified set of keyword assignments theoretically offers superior computational efficiency. However, effective fine-tuning on limited, domain-specific datasets remains technically challenging as well as expensive, and accurate literature classification in this context necessitates a nuanced comprehension of complex physical phenomena, a capability that is inherently difficult to encode within a reduced-parameter architecture. The stringent accuracy requirements of this task are precisely what make the trade unattractive.

Beyond the classification task itself, the structured records this pipeline produces invite further use. Shared properties across publications, such as common atomic transitions or common experimental methodologies, form a natural graph in which hidden connections between papers could be identified; mapping the resulting metadata in time would show when the properties of particular elements were established and how they evolved. Integration into the curation workflow in real time, so that new publications are categorized as they appear rather than in batches, is a practical extension once the questions of consistency raised in Sec. 3 are settled.

Finally, while the present work is focused on the bibliographic data, the next important step is the extraction and validation of the data presented in the text, tables, and graphs in the papers. In the subsequent step, a critical analysis of the entire set of available information should be performed. LLM-based systems to address these tasks are being developed and will be presented elsewhere.

5 Data and Code Availability

The pipeline, the topic prompts in their versioned form, and the evaluation scripts used to produce Tables 1 and 2 are available at `https://github.com/yralchen1/BibClass`. Generation was performed with a general-purpose large language model at temperature zero; the pipeline is not tied to a particular model or vendor. The scoring scripts require no model access, so all reported metrics can be recomputed from the stored keyword records.

Acknowledgments

The work of A.K. and Yu.R. was supported by NASA under award number 80GSFC24M0006.

Author Contributions

All authors contributed to conceptualizing the project and writing, editing, and proofreading the manuscript. M.P. and A.K. performed LLM engineering and Python coding. A.K., M.P. and Y.R. designed and implemented the evaluation suite and produced the benchmark results. A.K. and Y.R. prepared the validation sets.

References

[1] A. Kramida et al. NIST Atomic Spectra Database (ver. 5.12), [Online]. Available: https://physics.nist.gov/asd [2026, March 16]. National Institute of Standards and Technology, Gaithersburg, MD. 2024.

[2] J. R. Fuhr and W. L. Wiese. "Bibliography on atomic transition probabilities, July 1971 through June 1973". In: *NBS Special Publication 320, Suppl. 2*. U.S.: Nat. Bur. Stand., 1973. URL: https://www.govinfo.gov/app/details/GOVPUB-C13-0e024d314a3d917630978dafe02bddf9.

[3] Donald Norman Blake Hall. *An atlas of infrared spectra of the solar photosphere and of sunspot umbrae, in the spectral intervals 4040 cm^{-1}–5095 cm^{-1}; 5550 cm^{-1}–6700 cm^{-1}; 7400 cm^{-1}–8790 cm^{-1}*. Tucson, AZ, USA: Kitt Peak National Solar Observatory, 1973, 811 pp.

[4] C. E. Moore. *Bibliography on the analyses of optical atomic spectra – Sect. 1: ^{1}H–^{23}V*. U.S.: Nat. Bur. Stand., 1968. DOI: 10.6028/NBS.SP.306.

[5] Manas Pandey et al. "Solving tricky quantum optics problems with assistance from large language models". In: *Communications Physics* 8.1 (2025), p. 428. DOI: 10.1038/s42005-025-02398-2.