Searching Four-Millenia-Old Digitized Documents: A Text Retrieval System for Egyptologists

Estíbaliz Iglesias-Franjo and **Jesús Vilares**
Grupo Lengua Y Sociedad de la Información (LYS), Departamento de Computación
Facultade de Informática, Universidade da Coruña
Campus de A Coruña, 15071 – A Coruña (Spain)
{estibaliz.ifranjo, jesus.vilares}@udc.es

Abstract

Progress made in recent years has led to a growing interest in Digital Heritage. This article focuses on Egyptology and, more specifically, the study and preservation of ancient Egyptian scripts. We present a Text Retrieval system developed specifically to work with hieroglyphic texts. We intend to make it freely available to the research community. To the best of our knowledge this is the first tool of its kind.

1 Introduction

Until recently, the development of Information Retrieval (IR) systems has mainly focused on contemporary languages. From a socio-economic point of view, this makes perfect sense since our needs, as users, are connected to our everyday tasks, which we develop in our languages. Why should we pay attention to dead languages such as Ancient Egyptian? Our civilization was born in Mesopotamia and Egypt, and the culture of Pharaohs has fascinated us for decades and even centuries. Even nowadays, Egyptology continues to be one of the major branches of Archaeology and it is not unusual to find, from time to time, that new discoveries in this field open our news bulletins. Moreover, Egyptian is the longest-attested language, it thus becoming a particularly valuable object of research for Diachronic Linguists (Loprieno, 1995). However, neither should we forget its intrinsic value as one of the most representative elements of one of the most important human civilizations of all time. Egyptian Hieroglyphic script is a major component of our cultural heritage and, for that very reason, we should put particular emphasis on its preservation and study.

At this point, we need to introduce *Digital Heritage*, the scientific area that focuses on the use of computing and information technologies for the preservation and study of the human cultural legacy for current and future generations.

In this context, this work describes an open source Text Information Retrieval (TIR) system designed specifically for the processing of Egyptian Hieroglyphic scripts. To the best of our knowledge this is the first tool of its kind.

The rest of the paper is structured as follows. Firstly, Section 2 makes an introduction to Ancient Egyptian. Secondly, Section 3 describes how to encode hieroglyphic texts. Previous related work is outlined in Section 4. Next, the requirements of our system are analysed in Section 5, which is then described in Section 6. Finally, Section 7 presents our contributions and future work.

2 Language and Writing System

2.1 History

As previously commented, Egyptian (Allen, 2014; Loprieno, 1995; Cervelló-Autuori, 2015) is the longest-attested human language, with a documented history that spans several millenia, from about 3300 BC until the present day, when it is still used by the Coptic Christian Church in its rituals. Of course, it has undergone profund changes throughout its lifetime. So, we can distinguish two main phases in its development: *Earlier Egyptian*, whose writing system corresponds to the stereotypical image we have of Egyptian and that lasted as a spoken language from its origins until after 1300 BC; and *Later Egyptian*, which started to be used at that time and, after continuous evolution, survived until the 11th century AD as a productive language and until today as the ritual language of the Coptic Church. Our work focuses on Earlier Egyptian because of its archaelogical interest, in particular in the so-called *Middle or Classic Egyptian*, which remained as the traditional language

Proceedings of the 10th SIGHUM Workshop on Language Technology for Cultural Heritage, Social Sciences, and Humanities (LaTeCH), pages 22–31,
Berlin, Germany, August 11, 2016. ©2016 Association for Computational Linguistics

of hieroglyphic inscriptions until the fifth century AD, thus still being widely used in royal inscriptions, religious literature and monuments. From now on, unless we specify the contrary, we will be referring to *Middle Egyptian* when using the terms “Ancient Egyptian” or just “Egyptian” for short.

2.2 Characteristics

Egyptian belongs to the Afro-Asiatic language family, the same as contemporary languages such as Arabic, Hebrew and Berber, although Egyptian constitutes a subfamily of its own.

As in the case of early Arabic and Hebrew, Egyptian is a *consonantal* language since its words are formed from a consonantal root with vowels being used to indicate inflectional or derived forms. For the same reason, only consonants are written.

Its writing system is *pictographic* since its signs, or *hieroglyphs*, consist of symbols portraying beings and elements of the Egyptian world: parts of the human body (𓁹: an *eye*), plants (𓇋: a *reed*), animals (𓅭: a *pintail duck*), objects (𓊡: a *mast with sail*), etc.

It is also *logographic* since some, but not all, symbols have a meaning that corresponds, directly or indirectly (e.g. through a cultural, metonymic or metaphoric relation), to the same real-word element they reproduce. For example: 𓁹, an *eye* for *eye*; and 𓊡, a *mast with sail* for *wind*.

Egyptian writing system is *phonographic* too, since part of its signs depict sounds. For example, 𓐍 for the phoneme /χ/, transliterated as *ḫ*.

Finally, Ancient Egyptian had an *inherently “open”* writing system with no fixed alphabet. The number of available signs progressively increased from about 800 hieroglyphs in the Old Kingdom period to more than 5,000 in the Greco-Roman period. Moreover, new symbols and variants continue to be discovered when ancient texts are analyzed (Rosmorduc, 2003a).

2.3 Sign Types

In contrast with the formerly-held common belief that Egyptian writing system is a purely symbolic one, its script is mainly phonetical and combines different types of signs.

The first group are the *phonograms* or *phonetic signs*. In these signs the image carries no meaning whatsoever, being used by convention to represent the sounds of language. We can distinguish three types of phonograms according to the number of consonantal sounds represented, from one to three: *uniliterals*, e.g. 𓐍 (*ḫ*),[1]; *biliterals*, e.g. 𓅭 (*sꜣ*); and *triliterals*, e.g. 𓆣 (*ḫpr*).

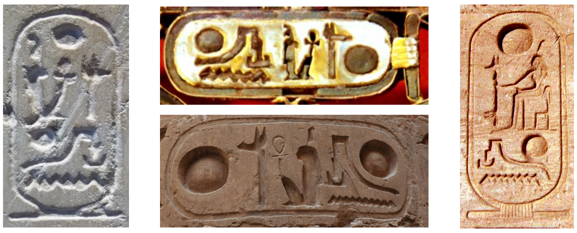


Figure 1: The four possible ways of writing the prenomen of Ramesses II by varying its direction.

The other group are the *semagrams*. In this case, the image of the sign participates directly in the codification and the significance of the linguistic message. In turn, we can distinguish two types of semagrams. Firstly, the *ideograms* (aka *logograms*) or *lexical signs*. They represent the things they actually depict and, consequently, are read that way. For example 𓁹, that depicts an eye and represents the word *ỉrt*, which means “*eye*”; and 𓏞, which depicts a scribe’s kit and is read *sḫꜣ*, used for “*write*” and related words. The second type are the *determinatives* or *semantic signs*. These signs are placed at the end of a word to indicate that it corresponds to a given semantic group. They are of great importance since they allow the reader to differentiate between words that have the same consonantal representation but different meaning. Unlike ideograms, determinatives are silent so they are not read. As an example, given the above-mentioned ideogram 𓏞, and the determinatives 𓏛 (category [WRITING - ABSTRACT NOTIONS]) and 𓀀 (category [MAN - HUMAN BEING]), the word 𓏞𓏛 means “*to write*” while the word 𓏞𓀀 means “*scribe*”.

It should be noted that the same glyph may belong to more than one category at once. For example, depending on the context, 𓈗 can be interpreted as the biliteral phonogram *mw*, the ideogram *mw* (which means “*water*”) or the determinative [WATER - LIQUIDS].

2.4 Writing Direction

Egyptian writing system is very flexible with regard to its *direction of writing*, which is not fixed.

[1] Where appropriate we will indicate the transliteration corresponding to the hieroglyphic text in question.

Hieroglyphic texts can be found written in horizontal rows, as with English and Arabic, or in vertical columns, as with traditional Japanese, Chinese and Mongolian. Moreover, although they are always read from top to bottom, they may follow a left-to-right ordering, as with English and Mongolian, or a right-to-left ordering, as with Arabic and Japanese. The reason for such a variety comes from the fact that Egyptian hieroglyphic script had a marked artistic nature (Cervelló-Autuori, 2015). It was intended to be carved or painted in monuments, walls, jewels, etc., even taking part of the scene itself (Rosmorduc, 2003a), and since one of the main characteristics of Egyptian art was its symmetry, they required their writing to adapt to it. Figure 1 presents a good example of its variety.

2.5 Sign Arrangement

Another remarkable feature is *continuous writing*, in which all the words run together with no dividers to separate words or phrases. This is also characteristic of some contemporary languages such as Chinese or Japanese, where no word separators are used. For example, in the case of the text 𓇋𓅱𓄿𓊪𓂧𓅱𓅭𓏥𓁷𓈖𓉔𓏏𓆭 (*iw 3pdw ḥr nht*), it stands for "*The birds are on the sycamore*".

Additionally, hieroglyphs were not arranged one after the other, in a linear way, as in the case of our writing system. Instead, scribes gathered them in so-called *groups*, trying to fill the space available neatly, in a way which resembles contemporary Hangul Korean script. Thus, as shown above, "*sycamore*" was not written 𓈖𓉔𓏏𓆭 (*nht*), but 𓈖𓉔𓏏𓆭 instead.

This arrangement depended, of course, on the words to be written, but also on several principles or heuristics (Cervelló-Autuori, 2015) the scribe followed in order to obtain the most harmonious and aesthetic arrangement possible.

3 Encoding Hieroglyphic Texts

Egyptologists and Linguists needed a practical way to represent hieroglyphic texts without having to re-draw their signs. The problem was solved by using regular characters to encode those texts.

3.1 Gardiner's List and the Extended Library

Named after its creator, the Egyptologist Sir Alan Gardiner (1957), *Gardiner's List*, a standard reference in the study of Egyptian, classifies its signs into 26 categories according to their drawing, each one identified with a letter: category `A` corresponds to "*Man and his occupations*" (𓀀𓀁𓀂. . .); `B` to "*Woman and her occupations*" (𓁐𓁑𓁒. . .); etc. In turn, hieroglyphs within each category are numbered sequentially so a given sign can be coded using the letter of its category and its corresponding number. For example, the code `E8` corresponds to the sign 𓃙 ("*goat kid*"), the eigth element of category `E` ("*Mammals*"). This classification includes the most common hieroglyphs (743 signs and 20 variants), enabling us to encode a significant proportion of the texts.

Symbol	Operation	Example	
-	concatenation	`Q3-X1-Z4-N1`	𓊪𓏏𓏭𓇯
:	subordination	`X1:Z4:N1`	𓏏𓏭𓇯
*	juxtaposition	`Q3*X1:Z4`	𓊪𓏏𓏭
()	grouping	`Q3*(X1:Z4):N1`	𓊪𓏏𓏭𓇯

Table 1: Sign arrangement operators in MdC.

In the 1990s, this list was largely extended to include newly identified signs and variants, thus becoming the so-called *Extended Library* (Grimal et al., 2000), with 4706 symbols. Gardiner's classification was not modified since new signs were numbered after the existing ones, and variants of existing signs were codified by attaching an extra letter to its code. For example, the symbol 𓃚 (code `E8a`) was added as variant of 𓃙 (code `E8`).

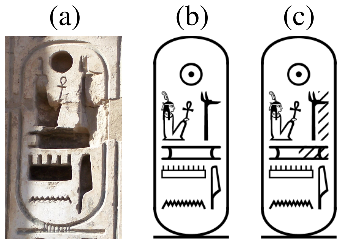
3.2 Manuel de Codage and its Dialects

In the 1980s, the *International Association of Egyptologists (IAE)*[2] formed a committee with the aim of developing a standard encoding system for the digitalization of hieroglyphic texts. The resulting document was the *Manuel de Codage (MdC)* (Buurman et al., 1988), an evolution of *Gardiner's List* (later adapted to the *Extended Library*) where new codes and rules were added for the accurate representation of hieroglyphs and other features of Egyptian writing system by using ASCII text. Next, we introduce an overview of the most significant additions.

3.2.1 Sign Operators

Table 1 shows, in order of precedence, the basic operators for arranging the signs. Thus, returning to our previous example, 𓈖𓉔𓏏𓆭 ("*sycamore*") is `N35:O4*X1-M1`.

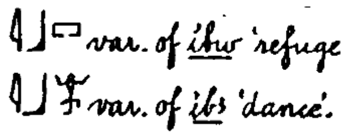
[2] `http://www.iae-egyptology.org/`



(b) `<-N5-F12*C10-N36-M17*(Y5:N35)->`

(c) `<-N5-(F12#13)*C10-N36#13-M17*(Y5:N35)->`

Figure 2: *(a)* Photo of a damaged cartouche showing the prenomem of Pharaoh Ramesses III; *(b)* MdC code corresponding to the undamaged cartouche and the output obtained from it with JSESH; and *(c)* MdC code corresponding to the shaded cartouche and its corresponding output.



var. of ỉḥw 'refuge'.

var. of ỉb3 'dance'.

Figure 3: Example of a handwritten entry, then printed lithographically, from Faulkner (2006).

3.2.2 Damaged Texts

The majority of the hieroglyphics that have survived until the present day have suffered the effects of time, exposure, vandalism, etc. So, one of the specific problems to be faced in this context was the representation of these texts in the most informative way. This matter was solved by the use of *shades*, implemented as special marks attached to the sign codes and which allow us to express whether the sign or even its presence is recognizable or not, how many signs are affected, which parts of them are damaged, etc. Figure 2 shows a simple example of their use.

3.2.3 Non-Hieroglyphic Text

MdC includes encoding support for combining hieroglyphs, transliterations, translations and other types of annotation within the same text. It assumes that all text is hieroglyphic unless it is enclosed between a given set of marks; for example '`+l`' (opening) and '`+s`' (closing) for enclosing regular text encoded in Latin script.

3.2.4 Dialects

Although the MdC should have been taken as the encoding standard for hieroglyphic text editors (see Section 4), the developers of these systems instead established their own particular specifications taking the MdC as their base, thus giving birth to different *dialects*. This meant that, in practice, with a few exceptions, a text written with a given program can not be opened and edited with another one unless it has been previously rewritten in the new notation. This fact not only makes it difficult to share documents between researchers and establish common corpora (Gozzoli, 2013), but also decreases the lifespan of those dialects and their encoded documents because of their dependence on that particular software they were created with and the fonts they use (Nederhof, 2013).

3.3 Unicode

As stated by Mark-Jan Nederhof (2013), the case of the inclusion of Egyptian Hieroglyphs in Unicode is a very good illustration of the troubles derived from trying to adapt other writing systems to Egyptian and its peculiarities. The process took more than a decade from the first proposal to its inclusion in Unicode 5.2. The list of available signs contains 1071 hieroglyphs (range U+13000..U+1342F) including the original Gardiner's List, its supplements and some other symbols (Everson and Richmond, 2007). Unfortunately, Unicode hieroglyphs encoding is limited by the lack of important features such as the availability of shading mechanisms, sign grouping or varied writing directionality (Richmond, 2015), thus making it a non-practical choice for many tasks.

3.4 Revised Encoding Scheme

Seeking to solve the current limitations of MdC, the above-mentioned software- and font-dependence of its dialects, and the formatting limitations of Unicode hieroglyphs, Mark-Jan Nederhof (2013) proposed the so-called *Revised Encoding Scheme (RES)*, which lacks such dependences and includes new sign operators. Although it requires more sophisticated processing than the MdC because of its added complexity, future hieroglyphic text processing systems will be probably influenced by this new scheme (Rosmorduc, 2015).

4 Related Work

The research community working on the application of Computer Science to Egyptology is small (Polis et al., 2013b). In the case of the computer processing of hieroglyphic text, it has been closely linked to the development of classic-style text editors (Gozzoli, 2013; Diop, 1992; Grimal, 1990). Since there were no hieroglyphic typewriters, scholars had to rely on handwritten texts when writing and sharing documents, a practical limitation that could easily lead to misinterpretations. Even in the case of books, the hieroglyphic texts printed in their pages were very complex and costly typographical transcriptions or, most of the time, mere lithographical copies of those handwritten by their authors, as shown in Figure 3, for example. Thus, the need for hieroglyphic text processing software was peremptory.

Among the specialized, and scarce, text processor software developed for this purpose, we should highlight two tools in particular. Firstly, GLYPH (Gozzoli, 2013), developed by Jan Buurman, which laid the foundations of future hieroglyphic text processors. It was published for DOS in 1986 and subsequently evolved and migrated to other operating systems: MACSCRIBE for Macintosh and WINGLYPH for Windows (3.1 and 95). The second tool we want to cite is JSESH, developed by Serge Rosmorduc (2014), which is, currently and in all probability, the most widely used word processor in Egyptology.

With regard to Text Mining and Natural Language Processing (NLP), Egyptian is, basically, a virgin territory waiting to be explored, namely because of the lack of computer corpora to work with (Rosmorduc, 2015). The reason for it is that hieroglyphic encoding is very time-consuming (Rosmorduc, 2015; Nederhof, 2015). However, those advances recently made in projects *Thesaurus Linguae Aegyptiae (TLA)* (Dils and Feder, 2013) and *Ramsès* (Polis et al., 2013a; Polis and Rosmorduc, 2013) are promising. Anyway, a few works about automatic transliterion (Barthélemy and Rosmorduc, 2011), language modeling (Nederhof and Rahman, 2015a) and text categorization (Gohy et al., 2013) can be found.

Recent advances in Egyptian OCR are of interest (Franken and van Gemert, 2013; Nederhof, 2015), since OCR would greatly reduce the cost of encoding these texts (Piotrowski, 2012, Ch. 4).

5 Requirements of the System

Our goal has been to develop an IR system capable of operating on Egyptian texts. For this purpose, we have studied the nature of this language and its writing system, and consulted an expert Egyptologist to better understand the application domain. As a result, we established the following requirements:

1. **Simplicity**: It should be intuitive and easy to use, with a minimum learning curve.
2. **Content indexing**: The system must be able to index documents containing conventional text and hieroglyphic text. At first we will focus on those documents written with JSESH, thus covering a significant proportion of the digitalized contents currently available.
3. **Querying using MdC encoding**: In the case of hieroglyphs, users will input the query using MdC encoding, with which they are already familiarized.
4. **Display the query using glyphs**: In order to make it easier for the user, the system will display, in parallel, the input MdC query using pictograms.
5. **Querying using conventional text**: Since the documents contain both hieroglyphic and conventional text (encoded in Latin script), we also want to be able to submit conventional text queries.
6. **Submission of mixed queries**: The possibility of making queries combining both hieroglyphic and conventional text.
7. **Relevant documents retrieval**.
8. **Display of document contents**: The user should be able to access the content of the documents retrieved by the system and check why they have been retrieved.

6 Description of the System

The architecture of our IR system, currently available under a free license at `http://github.com/estibalizifranjo/hieroglyphs`, corresponds, in general, to a classic Text Retrieval system, as shown in Figure 4. Two main phases of functioning can be distinguished: firstly, the indexing of the document collection on which searches are to be performed and, secondly, the querying–retrieval process. Next, we describe those modules of the system involved in each of these phases.

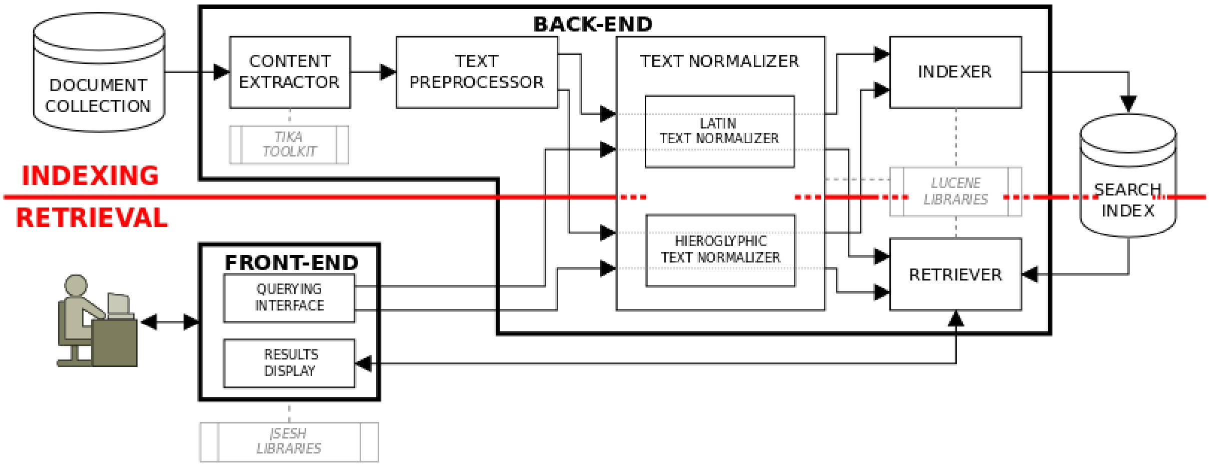


Figure 4: Schematic representation of the system: indexing and retrieval processes.

6.1 Phase 1: Indexing

It consists of extracting and indexing the content of the documents on which searches will be performed later.

6.1.1 Content Extraction

This module uses the Tika toolkit[3], which can detect and extract both text and metadata from a wide range of different file types (ODT, DOC, PDF, etc.), to extract the text of the documents.

6.1.2 Text Preprocessing

The obtained text is then preprocessed to separate conventional text from hieroglyphic text and to filter out irrelevant data. For this task the system applies a *pattern matching* approach. For instance, in the case of detecting pieces of unformatted conventional text, it uses a regular expression for identifying sequences of characters enclosed between the marks '`+l`' and '`+s`', corresponding to regular unformatted text, as explained in Section 3.2.3.

6.1.3 Conventional Text Normalization

The normalization components apply a series of *text operations* for tokenizing, conflating and generating the index terms of the input texts. The nature of such operations varies according to the type of text: regular text or hieroglyphs. For its implementation we have taken as our basis Apache Lucene.[4] In the case of conventional text, a standard processing is performed (Manning et al., 2008): firstly, a standard lexical analysis is applied for tokenizing the text, and the resulting terms are then conflated by lowercasing them and removing both stopwords and diacritics.

6.1.4 Hieroglyphic Text Normalization

Due to its peculiarities, hieroglyphic text is processed in a completely different way. The first problem is the lack of delimiters to separate words or phrases. Although MdC provides special markers for this purpose, in practice they are not used since they have no effect on the text graphical representation. As an initial solution, we have used *sign groups* (Section 2.5) as a working unit since they are delimited by '`-`' at encoding level. For example, the word 𓈖𓉔𓏏𓆭 (`N35:O4*X1-M1`) is composed of four signs but only two groups, so it would be tokenized into 𓈖𓉔𓏏 (`N35:O4*X1`) and 𓆭 (`M1`). This time input text will not be lowercased, since MdC encoding is case-sensitive, neither the punctuation marks will be removed, since they form part of MdC encoding.

6.1.5 Index Generation

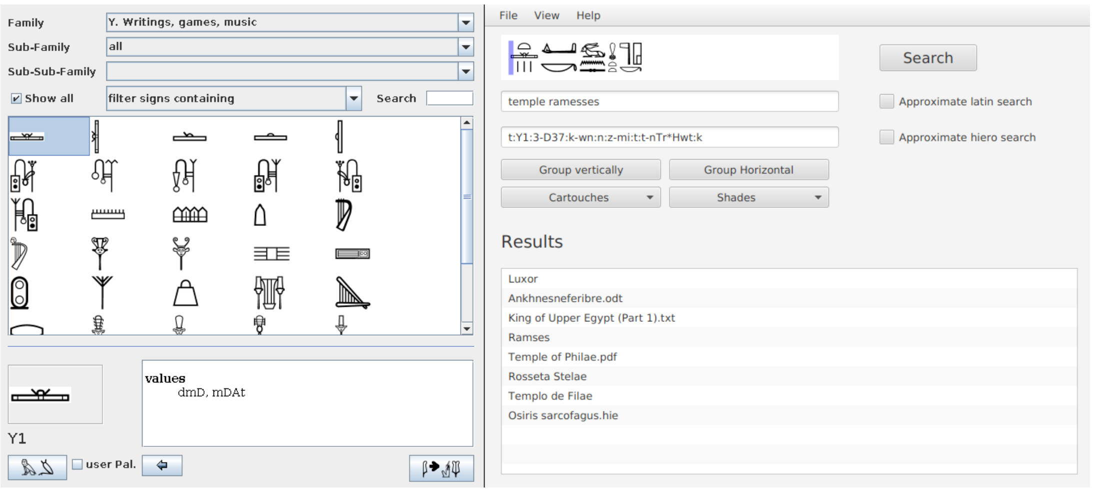
Finally, the index structure is generated. In the case of the hieroglyphic text, the sign groups are indexed together with their occurrence positions within the text. This module has also been implemented using Lucene.

6.2 Phase 2: Querying–Retrieval

Two main sub-processes can be distinguished in this second phase, the querying process and the retrieval process, which can be controlled through the front-end interface of the system.

[3] `http://tika.apache.org`

[4] `http://lucene.apache.org/core/`



Family: Y. Writings, games, music

Sub-Family: all

Sub-Sub-Family:

☑ Show all

filter signs containing

values
dmD, mDAt

Y1

□ user Pal.

temple ramesses

t:Y1:3-D37:k-wn:n:z-mi:t:t-nTr*Hwt:k

□ Approximate latin search

□ Approximate hiero search

Results

- Luxor
- Ankhnesneferibre.odt
- King of Upper Egypt (Part 1).txt
- Ramses
- Temple of Philae.pdf
- Rosseta Stelae
- Templo de Filae
- Osiris sarcofagus.hie

Figure 5: Screenshot of the front-end querying interface. A mixed query containing both Latin and hieroglyphic text (*top of right-hand panel*) has been composed, the latter with the assistance of the symbol palette (*left-hand panel*). The list of relevant documents retrieved by the system is already available (*bottom of right-hand panel*).

6.2.1 Querying

The user can query the indexed collection by using either hieroglyphics, regular text (in the Latin script) or a combination of both (*mixed* queries), that is, a query containing both hieroglyphic text and conventional text at the same time, such as that one shown in Figure 5, for example. The query normalization process is parallel to that performed during the indexing. In the case of hieroglyphic text, the *exact matching* mode requires the documents to contain exactly the same group sequence specified in the query (i.e. the same signs with the same arrangement), while the *approximate matching* mode allows the user to sub-specify the composition of a group (e.g. to require that a given group of the sequence contains a given sign but without specifying whether it contains any more symbols or their arrangement within the group).

6.2.2 Retrieval

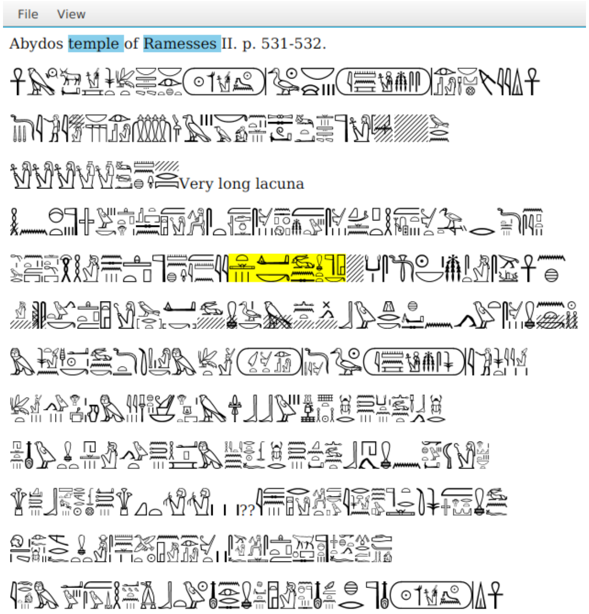
Once the query has been normalized, the recovery module accesses the index looking for matches and identifies those documents of the collection that are relevant to the query. The current implementation combines two retrieval models (Manning et al., 2008): firstly, the relevant documents are selected by using a Boolean model and, then, a Vector Space model is used to score and rank those previously selected documents. The resulting document list will be returned and presented to the user.

6.2.3 Front-End Interface

Particular attention has been paid to the design of the interface to make its use as easy and intuitive as possible. As shown in the top of the right-hand panel of Figure 5, separate search forms are provided for conventional text (in the Latin script) and hieroglyphic text queries. In the case of the latter, those pictograms corresponding to the MdC code text being introduced will be automatically displayed so that the user can check them on the fly.

At this point, we decided to integrate additional features not considered in the original requirements, in order to improve the usability and flexibility of the interface. Following the example of the JSESH editing tool, our interface provides users, if required, with a palette of hieroglyphic signs that enables them to add symbols to the query by clicking on them, as shown in the left-hand panel of Figure 5. This palette also functions as a catalog of symbols organized according to Gardiner's List classification (Section 3.1), so the user can navigate through it and consult the information and variants associated with each symbol. The interface also provides several options for handling the hieroglyphic text, such as adding shadows or creating personalised palettes.

In the case of hieroglyphic queries, another possible choice for its input would have been to use a similar approach to that one proposed by Tetsuo Minohara (2010), which is based on the



Abydos temple of Ramesses II. p. 531-532.

Very long lacuna

Figure 6: Content of one of the documents retrieved for our sample query, as presented by the system interface, which is highlighting the matchings found during the retrieval process.

Japanese Kanji writing method. However, this approach, although interesting, was not intuitive and too complex for a non-Japanese user.

At the same time, the interface is also responsible for presenting the user with the result of the search, as shown in the bottom of the right-hand panel of Figure 5. Moreover, it enables the user to access the content of these documents, which, if so required, will be displayed highlighting the matchings found during the retrieval process, as can be seen in Figure 6. Thus, the system provides the user with useful feedback about why the document has been retrieved.

For its implementation we have made use of the libraries provided with JSESH, including its symbol palette. This was intentional since, as previously explained in Section 4, JSESH is, currently, the most popular editing tool among the Egyptology community. This way, novice users of our system will find an interface with a very similar appearance and behavior to that of the editing tool they are already familiar with, thus greatly facilitating its use and minimizing the learning curve.

7 Conclusions and Future work

Ancient Egyptian Text Mining is still in the initial stages of development. We have presented in this work a Text Information Retrieval system specifically designed to manage Egyptian hieroglyphic texts which, to the best of our knowledge, is the first tool of its kind. For its development we have taken into account the lexical and encoding characteristics of this language and its writing system. Apart from the conflation process to be applied in the case of the Egyptian text, we have taken special care with the design of the front-end interface in order to make it as intuitive and easy to use as possible for novel users, paying particular attention to the case of Egyptologists, its intended future users. Our first distribution have been released under a free license.

We intend to continue adding new features to the system. New input filters, for example, would allow the system to extend the range of source document types accepted as input: documents created with other hieroglyphic text editors, Unicode hieroglyphic text or, as in the case of this article, HieroTEX LATEX documents (Rosmorduc, 2003b).

From an IR perspective, we would like to continue studying how to improve performance. One possible choice is the application of a more flexible retrieval solution using a single retrieval model instead of the current double-model 2-stage retrieval process. Classic Vector Space and Probabilistic models (Manning et al., 2008) are the first options. However, the very special and noisy nature of Egyptian writing system and the application context may suggest the use of other approaches: the use of standard character n-grams as a working unit, a solution successfully applied in both noisy contexts (Vilares et al., 2011) and languages whose writing systems share characteristics with Egyptian, such as Japanese (Ogawa and Matsuda, 1999), Chinese (Foo and Li, 2004), Korean (Lee and Ahn, 1996) or Arabic (Mustafa and Al-Radaideh, 2004); the use of so-called character *s-grams* (Järvelin et al., 2008), a generalization of the concept of n-gram by allowing *skips* during the matching process; the application of locality-based models (de Kretser and Moffat, 1999); or phonetic matching (Yasukawa et al., 2012). Closer to the NLP field, the development of *conflation mechanisms* based on lemmatization or morphological analysis (Piotrowski, 2012, Ch. 7) would be very useful. However, many of these solutions would require a further study of the language and its writing system, and the development of resources such as evaluation corpora, which were beyond the scope of this initial project, although we intend to contact, in a close future, experts in the field to try to solve these questions.

Acknowledgments

This research has been partially funded by the Spanish Ministry of Economy and Competitiveness (MINECO) through project FFI2014-51978-C2-2-R. We would like to thank Dr. Josep Cervelló Autuori, Director of the Institut d'Estudis del Pròxim Orient Antic (IEPOA) of the Universitat Autònoma de Barcelona for introducing us to the Ancient Egyptian script and acting as our fictional client. We would also like to thank Dr. Serge Rosmorduc, Associate of the Conservatoire National des Arts et Métiers (CNAM) for all his support when working with JSESH.

References

James P. Allen. 2014. *Middle Egyptian: An Introduction to the Language and Culture of Hieroglyphs (3rd Edition)*. Cambridge University Press.

François Barthélemy and Serge Rosmorduc. 2011. Intersection of multitape transducers vs. cascade of binary transducers: The example of Egyptian hieroglyphs transliteration. In *Proc. of the 9th International Workshop on Finite State Methods and Natural Language Processing (FSMNLP 2011)*, pp. 74–82. ACL.

Jan Buurman, Nicolas-Christophe Grimal, Michael Hainsworth, Jochen Hallof, and Dirk van der Plas. 1988. *Inventaire des signes hiéroglyphiques en vue de leur saisie informatique: manuel de codage des textes hiéroglyphiques en vue de leur saisie sur ordinateur*, volume 8 of *Mémoires de l'Académie des Inscriptions et Belles-Lettres*. De Boccard, Paris.

Josep Cervelló-Autuori. 2015. *Escrituras, Lengua y Cultura en el Antiguo Egipto*. El espejo y la lámpara. Edicions UAB.

Owen de Kretser and Alistair Moffat. 1999. Effective document presentation with a locality-based similarity heuristic. In *Proc. of 22nd Annual International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR'99)*, pp. 113–120. ACM Press. DOI 10.1145/312624.312664.

Peter Dils and Frank Feder, 2013. In (Polis et al., 2013b), chapter The Thesaurus Linguae Aegyptiae. Review and Perspectives, pp. 11–23. Project website: `http://aaew.bbaw.de/tla/` (visited on May 2016).

Cheikh M'Backé Diop. 1992. Hiéroglyphes et informatique. *ANKH: Revue d'Egyptologie et des Civilisations Africaines*, (1):105–121, February.

Michael Everson and Bob Richmond. 2007. Proposal to encode Egyptian Hieroglyphs in the SMP of the UCS. Working Group Document ISO/IEC JTC1/SC2/WG2 N3237 [L2/07-097]. Technical report, UTC (Unicode Technical Committee), Unicode Consortium, April. Unicode 8.0 hieroglyphs table available at: `http://www.unicode.org/charts/PDF/U13000.pdf` (visited on May 2016).

Raymond Oliver Faulkner. 2006. *Concise Dictionary of Middle Egyptian*. Griffith Institute.

Schubert Foo and Hui Li. 2004. Chinese word segmentation and its effect on Information Retrieval. *Information Processing and Management*, 40(1):161–190.

Morris Franken and Jan C. van Gemert. 2013. Automatic Egyptian hieroglyph recognition by retrieving images as texts. In *Proc. of the 21st ACM International Conference on Multimedia (MM'13)*, pp. 765–768. ACM.

Alan Henderson Gardiner. 1957. *Egyptian grammar: being an introduction to the study of hieroglyphs*. Griffith Institute, Ashmolean Museum, Oxford, 3rd ed., revised edition. A complete on-line version is available at: `http://en.wikipedia.org/wiki/Gardiner′s_sign_list` (visited on May 2016).

Stéphanie Gohy, Benjamin Martin Leon, and Stéphane Polis, 2013. In (Polis et al., 2013b), chapter Automated text categorization in a dead language. The detection of genres in Late Egyptian, pp. 61–74.

Roberto Gozzoli, 2013. In (Polis et al., 2013b), chapter Hieroglyphic Text Processors, Manuel de Codage, Unicode and Lexicography, pp. 89–101.

Nicolas Grimal, Jochen Hallof, and Dirk van der Plas. 2000. *HIEROGLYPHICA: Sign List– Liste des Signes – Zeichenliste (2nd Edition)*, volume 1ˆ2. Publications Interuniversitaires de Recherches Égyptologiques Informatisées, Utrecht–Paris. Second edition revised and enlarged by Jochen Hallof, Hans van den Berg and Gabriele Hallof. Online list available on: `http://hieroglyphes.pagesperso-orange.fr/CCER-Hieroglyphica.htm` (visited on May 2016).

Nicolas Grimal. 1990. Hiéroglyphes et ordinateurs. *BRISES. Bulletin de Recherches sur l'Information en Sciences Économiques Humaines et Sociales*, (15):57–60.

Antti Järvelin, Tuomas Talvensaari, and Anni Järvelin. 2008. Data driven methods for improving mono- and cross-lingual IR performance in noisy environments. In *Proc. of the Second Workshop on Analytics for Noisy Unstructured Text Data (AND'08)*, volume 303 of *ACM International Conference Proceeding Series*, pp. 75–82. ACM.

Joo Ho Lee and Jeong Soo Ahn. 1996. Using n-grams for Korean text retrieval. In *Proc. of the 19th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR'96)*, pp. 216–224. ACM.

Antonio Loprieno. 1995. *Ancient Egyptian: A Linguistic Introduction*. Cambridge University Press.

Christopher D. Manning, Prabhakar Raghavan, and Hinrich Schütze. 2008. *Introduction to Information Retrieval*. Cambridge University Press.

Tatsuo Minohara. 2010. A writing system for the Ancient Egyptian hieroglyphs. In *Proc. of the 7th International Conference on Informatics and Systems (INFOS 2010)*, pp. 1–7. IEEE.

Suleiman H. Mustafa and Qasem A. Al-Radaideh. 2004. Using n-grams for Arabic text searching. *Journal of the American Society for Information Science and Technology (JASIST)*, 55(11):1002–1007.

Mark-Jan Nederhof and Fahrurrozi Rahman. 2015a. A probabilistic model of Ancient Egyptian writing. In *Proc. of the 12th International Conference on Finite State Methods and Natural Language Processing (FSMNLP 2015)*. ACL.

Mark-Jan Nederhof, 2013. In (Polis et al., 2013b), chapter The Manuel de Codage Encoding of Hieroglyphs Impedes Development of Corpora, pp. 103–110.

Mark-Jan Nederhof. 2015. OCR of handwritten transcriptions of Ancient Egyptian hieroglyphic text. In *Altertumswissenschaften in a Digital Age: Egyptology, Papyrology and Beyond (DHEgypt15). Leipzig, Germany, November 4-6, 2015.*

Yasushi Ogawa and Toru Matsuda. 1999. Overlapping statistical segmentation for effective indexing of Japanese text. *Information Processing and Management*, 35(4):463–480.

Michael Piotrowski. 2012. *Natural Language Processing for Historical Texts*. Synthesis Lectures on Human Language Technologies. Morgan & Claypool Publishers.

Stéphane Polis and Serge Rosmorduc, 2013. In (Polis et al., 2013b), chapter Building a Construction-Based Treebank of Late Egyptian. The Syntactic Layer in Ramses, pp. 45–59.

Stéphane Polis, Anne-Claude Honnay, and Jean Winand, 2013a. In (Polis et al., 2013b), chapter Building an Annotated Corpus of Late Egyptian. The Ramses Project: Review and Perspectives, pp. 25–44. Project website: `http://www.egypto.ulg.ac.be/Ramses.htm`. Beta online system: `http://ramses.ulg.ac.be/` (both visited on May 2016).

Stéphane Polis, Jean Winand, and Todd Gillen, editors. 2013b. *Texts, Languages & Information Technology in Egyptology: Selected Papers from the Meeting of the Computer Working Group of the International Association of Egyptologists (Informatique & Égyptologie), Liège, 6-8 July 2010*, volume 9 of *Collection Ægyptiaca Leodiensia*. Presses Universitaires de Liège, Liège.

Bob Richmond. 2015. Egyptian Hieroglyphs in Unicode plain text: A note on a suggested approach [L2/15-069]. Technical report, UTC (Unicode Technical Committee), Unicode Consortium, February.

Serge Rosmorduc. 2003a. Codage informatique des langues anciennes. *Document numérique*, 6(3-4):211–224.

Serge Rosmorduc. 2003b. HieroTeX: A LaTeXperiment of hieroglyphic typesetting. Package available at: `http://www.ctan.org/tex-archive/language/hieroglyph` (visited on May 2016).

Serge Rosmorduc. 2014. JSESH documentation. Software available at: `http://jsesh.qenherkhopeshef.org/` (visited on May 2016).

Serge Rosmorduc. 2015. Computational linguistics in egyptology. In Julie Stauder-Porchet, Andréas Stauder, and Willeke Wendrich, editors, *UCLA Encyclopedia of Egyptology*. UCLA, Los Angeles, USA.

Jesús Vilares, Manuel Vilares, and Juan Otero. 2011. Managing Misspelled Queries in IR Applications. *Information Processing & Management*, 47(2):263–286.

Michiko Yasukawa, J. Shane Culpepper, and Falk Scholer. 2012. Phonetic matching in Japanese. In *Proc. of SIGIR 2012 Workshop on Open Source Information Retrieval (OSIR 2012)*, pages 68–71.

A Third-Party Pictures

Figure 1, left (cropped picture): original by *Khruner*; available in Wikipedia under the Creative Commons Attribution-Share Alike 3.0 Unported license. **Figure 1, top** (cropped picture): original by *Hans Ollermann*; available in Wikipedia under the Creative Commons Attribution 2.0 Generic license. **Figure 1, right and bottom** (cropped pictures): originals by *Francesco Gasparetti*; available in Wikipedia under the Creative Commons Attribution 2.0 Generic license. **Figure 2, left** (cropped picture): original by *Lord-of-the-Light*; available in Wikipedia under the Creative Commons Attribution-Share Alike 3.0 Unported license.